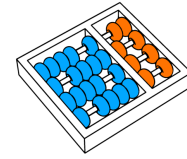


Fernando Granha Jeronimo

Quantum Computing: Automata, Games, and Complexity

Computação Quântica: Autômatos, Jogos e Complexidade

CAMPINAS
2015



University of Campinas
Institute of Computing

Universidade Estadual de Campinas
Instituto de Computação

Fernando Granha Jeronimo

Quantum Computing: Automata, Games, and Complexity

Supervisor: Prof. Dr. Arnaldo Vieira Moura
Orientador(a):

Computação Quântica: Autômatos, Jogos e Complexidade

MSC Thesis presented to the Graduate Program of the Institute of Computing of the University of Campinas to obtain a Mestre degree in Computer Science.

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Computação da Universidade Estadual de Campinas para obtenção do título de Mestre em Ciência da Computação.

THIS VOLUME CORRESPONDS TO THE FINAL VERSION OF THE THESIS DEFENDED BY FERNANDO GRANHA JERONIMO, UNDER THE SUPERVISION OF PROF. DR. ARNALDO VIEIRA MOURA.

ESTE EXEMPLAR CORRESPONDE À VERSÃO FINAL DA DISSERTAÇÃO DEFENDIDA POR FERNANDO GRANHA JERONIMO, SOB ORIENTAÇÃO DE PROF. DR. ARNALDO VIEIRA MOURA.

Supervisor's signature / *Assinatura do Orientador(a)*

CAMPINAS

2015

Ficha catalográfica
Universidade Estadual de Campinas
Biblioteca do Instituto de Matemática, Estatística e Computação Científica
Maria Fabiana Bezerra Muller - CRB 8/6162

J483q Jeronimo, Fernando Granha, 1987-
Quantum computing : automata, games, and complexity / Fernando Granha
Jeronimo. – Campinas, SP : [s.n.], 2015.

Orientador: Arnaldo Vieira Moura.
Dissertação (mestrado) – Universidade Estadual de Campinas, Instituto de
Computação.

1. Computação quântica. 2. Complexidade computacional. 3. Teoria dos
autômatos. 4. Teoria da computação. I. Moura, Arnaldo Vieira, 1950-. II.
Universidade Estadual de Campinas. Instituto de Computação. III. Título.

Informações para Biblioteca Digital

Título em outro idioma: Computação quântica : autômatos, jogos e complexidade

Palavras-chave em inglês:

Quantum computing

Computational complexity

Machine theory

Theory of computation

Área de concentração: Ciência da Computação

Titulação: Mestre em Ciência da Computação

Banca examinadora:

Arnaldo Vieira Moura [Orientador]

Franklin de Lima Marquezino

Julio Cesar López Hernández

Data de defesa: 17-08-2015


Programa de Pós-Graduação: Ciência da Computação

TERMO DE APROVAÇÃO

Defesa de Dissertação de Mestrado em Ciência da Computação, apresentada pelo(a)
Mestrando(a) **Fernando Granha Jeronimo**, aprovado(a) em **17 de agosto de 2015**, pela
Banca examinadora composta pelos Professores(as) Doutores(as):


Prof(a). Dr(a). Franklin de Lima Marquezino
Titular


Prof(a). Dr(a). Julio Cesar López Hernández
Titular


Prof(a). Dr(a). Arnaldo Vieira Moura
Presidente

Abstract

Since its inception, Theoretical Computer Science has dealt with models of computation primarily in a very abstract and mathematical way. The notion of efficient computation was investigated using these models mainly without seeking to understand the inherent capabilities and limitations of the actual physical world. In this regard, Quantum Computing represents a rupture with respect to this paradigm. Rooted on the postulates of Quantum Mechanics, it is able to attribute a precise physical notion to computation as far as our understanding of nature goes. These postulates give rise to fundamentally different properties one of which, namely entanglement, is of central importance to computation and information processing tasks. Entanglement captures a notion of correlation unique to quantum models. This quantum correlation can be stronger than any classical one, thus being at the heart of some quantum super-classical capabilities. In this thesis, we investigate entanglement from the perspective of quantum computational complexity. More precisely, we study a well known complexity class, defined in terms of proof verification, in which a verifier has access to multiple unentangled quantum proofs ($\text{QMA}(k)$). Assuming the proofs do not exhibit quantum correlations seems to be a non-trivial hypothesis, potentially making this class larger than the one in which only a single proof is given. Notwithstanding, finding tight complexity bounds for $\text{QMA}(k)$ has been a central open question in quantum complexity for over a decade. In this context, our contributions are threefold. Firstly, we study closely related classes showing how computational resources may affect its power in order to shed some light on $\text{QMA}(k)$ itself. Secondly, we establish a relationship between classical Probabilistically Checkable Proofs and $\text{QMA}(k)$ allowing us to recover known results in unified and simplified way, besides exposing the interplay between them. Thirdly, we show that some paths to settle this open question are obstructed by computational hardness. In a second moment, we turn our attention to restricted models of quantum computation, more specifically, quantum finite automata. A model known as Two-way Quantum Classical Finite Automaton (2QCFA) is the main object of our inquiry. Its study is intended to reveal the computational power provided by finite dimensional quantum memory. We extend this automaton with the capability of placing a finite number of markers in the input tape. For any number of markers, we show that this extension is more powerful than its classical deterministic and probabilistic analogues. Besides bringing advances to these two complementary lines of inquiry, this thesis also provides a vast exposition to both subjects: computational complexity and automata theory.

Resumo

Desde seu surgimento, Teoria da Computação tem lidado com modelos computacionais de maneira matemática e abstrata. A noção de computação eficiente foi investigada usando esses modelos sem procurar entender as capacidades e limitações inerentes ao mundo físico. A Computação Quântica representa uma ruptura com esse paradigma. Enraizada nos postulados da Mecânica Quântica, ela é capaz de atribuir um sentido físico preciso à computação segundo nosso melhor entendimento da natureza. Esses postulados dão origem a propriedades fundamentalmente diferentes, uma em especial, chamada emaranhamento, é de importância central para computação e processamento de informação. O emaranhamento captura uma noção de correlação que é única a modelos quânticos. Essas correlações quânticas podem ser mais fortes do que qualquer correlação clássica estando dessa forma no coração de algumas capacidades quânticas que vão além do clássico. Nessa dissertação, nós investigamos o emaranhamento da perspectiva da complexidade computacional quântica. Mais precisamente, nós estudamos uma classe bem conhecida, definida em termos de verificação de provas, em que um verificador tem acesso à múltiplas provas não emaranhadas ($\text{QMA}(k)$). Assumir que as provas não contêm correlações quânticas parece ser uma hipótese não trivial, potencialmente fazendo com que essa classe seja maior do que aquela em que há apenas uma prova. Contudo, encontrar cotas de complexidade justas para $\text{QMA}(k)$ permanece uma questão central sem resposta por mais de uma década. Nesse contexto, nossa contribuição é tripla. Primeiramente, estudamos classes relacionadas mostrando como alguns recursos computacionais podem afetar seu poder de forma a melhorar a compreensão a respeito da própria classe $\text{QMA}(k)$. Em seguida, estabelecemos uma relação entre Probabilistically Checkable Proofs (PCP) clássicos e $\text{QMA}(k)$. Isso nos permite recuperar resultados conhecidos de maneira unificada e simplificada. Para finalizar essa parte, mostramos que alguns caminhos para responder essa questão em aberto estão obstruídos por dificuldades computacionais. Em um segundo momento, voltamos nossa atenção para modelos restritos de computação quântica, mais especificamente, autômatos quânticos finitos. Um modelo conhecido como Two-way Quantum Classical Finite Automaton (2QCFA) é o objeto principal de nossa pesquisa. Seu estudo tem o intuito de revelar o poder computacional provido por memória quântica de dimensão finita. Nos estendemos esse autômato com a capacidade de colocar um número finito de marcadores na fita de entrada. Para qualquer número de marcadores, mostramos que essa extensão é mais poderosa do que seus análogos clássicos determinístico e probabilístico. Além de trazer avanços em duas linhas complementares de pesquisa, essa dissertação provê uma vasta exposição a ambos os campos: complexidade computacional e autômatos.

Contents

Abstract	vii
Resumo	viii
Acknowledgements	xiii
1 Introduction	1
2 Foundations	5
3 Ubiquity of Entanglement	31
4 Quantum Tools	51
5 Complexity Classes	71
6 Classical PCP	87
7 Variations of Unentangled Provers	103
8 PCPs and $\text{QMA}(k)$	119
9 Disentanglers and de Finetti	133
10 Area Law	145
11 On the Languages Recognized by 2QCFA	155
12 Quantum Marking and Multi-Head Automata	177
13 Conclusion	199
Bibliography	201

To my wife Renata.

Acknowledgements

I would like to thank the financial support provided by FAPESP¹.

¹This work was supported by FAPESP 2013/20661-1 and 2014/06467-0.

List of Figures

2.1	Quantum Computing	5
2.2	Bloch Sphere	16
3.1	CHSH States	42
3.2	CHSH Strategy	43
5.1	Interactive Proof System	78
5.2	QIP(3) Circuit	80
5.3	Refereed Game	81
5.4	MIP	82
5.5	Complexity Diagram	84
6.1	PCP Verifier	89
6.2	PCP Equivalences	90
6.3	Projection Constraint	97
7.1	Quantum Logarithm Interaction	107
10.1	2-D Grid	146
10.2	1-D Chain	146
10.3	Chebyshev Polynomial	151
11.1	2QCFA and The Chomsky Hierarchy	173

List of Tables

3.1	CHSH Successful States	42
3.2	Magic Square	45
3.3	Magic Square Strategy	47
12.1	Automata Summary	181

Chapter 1

Introduction

The study of quantum computing goes way beyond trying to use Quantum Mechanics to devise faster than classical algorithms. Quantum information is unique in its own right. Coherent superposition, interference, and entanglement are distinctive properties of this model. Quantum computing exploits our ultimate understanding of the world at small scales relying on the postulates of this theory to attribute a precise physical notion to computation. Contrary to the abstract model of a Turing Machine, in this new paradigm we are trying to understand the capabilities and limitations of computation in the physical world. In classical computability theory, the study of restricted models of computation such as finite automata is important to shed light on the capabilities and limitations of computational resources. In the quantum case, the power of quantum resources can also be investigated using restricted quantum models comprising generalizations of classical finite automata. In this context, computational complexity goes one step further by quantifying the resources necessary to accomplish computational tasks. This thesis embodies advances in two complementary lines of inquiry: the role of entanglement in quantum complexity and quantum finite automata. Furthermore, it provides an extensive exposition of both subjects aimed to equip the novice with a foundation necessary to start grasping state-of-the-art results in these fields.

Using Preskill's analogy, the information of a classical book is contained in its pages whereas a quantum book may contain information also in combinations of pages rather than only in individual pages. Extending his analogy we can say that after splitting a quantum book in two parts, it might not be possible to describe its entire content by the combined content of the parts. There might be states of the first part that depend on the second one. The phenomenon captured by this metaphor is the so called entanglement phenomenon. In this case, the two parts are said to be entangled with one another. Highly entangled states are widely believed to have no efficient classical representation[80]. Therefore, computations and information processing tasks involving this kind of state

might be at the heart of quantum superclassical capabilities. Computational complexity provides a lens to study sciences shedding light on their capabilities and limitations. In this work, entanglement is investigated from this perspective. More specifically, we used the important open question of finding tight bounds for the complexity class $\text{QMA}(k)$ as an inspiration to study entanglement.

The class QMA is a natural quantum generalization of the famous NP class in which the proof and the verifier are quantum. In turn, the class $\text{QMA}(k)$ is a generalization of QMA in which the verifier is promised to receive k unentangled proofs. There are surprising results indicating that unentanglement might be a non-trivial hypothesis making this extension more powerful than QMA. On the other hand, some restricted versions of $\text{QMA}(k)$ were shown to be in QMA. The inspirational question about tight bounds for $\text{QMA}(k)$ remains open. However, this work brings contributions in improving our understanding of this problem and in showing why some paths to settle this question are obstructed by computational hardness barriers.

To gain a better understanding about $\text{QMA}(k)$, it is useful to explore some closely related complexity classes. Allowing slight variations in the definition of $\text{QMA}(k)$ may clarify what properties can make it larger while others may keep it unchanged. Variations of unentanglement multi-prover quantum interactive system QMIP_{ne} were studied. Building on the result of Beige et al. [15], we showed that $\text{QMA}(k)$ remains the same even if the verifier is allowed to choose a constant number of provers to send logarithmic size queries. This kind of result allows more flexibility in designing protocols in $\text{QMA}(k)$. Moreover, the simple equivalence $\text{QMA}_{\log(n)}(\text{poly}) = \text{QCMA}$ from [43] is proved in a different way. This equivalence has the implication that $\text{QMA}_{\log(n)}(\text{poly})$ has perfect completeness. We present these results in chapter 7.

Using classical PCP results, we extend an existing $\text{QMA}(k)$ protocol for an NP-complete problem in a generic way, making explicit the dependence of the final completeness soundness gap on the various PCP parameters. This generic treatment allows us to recover known results about the relationship of $\text{QMA}(k)$ with NEXP and 3SAT in a unified and simplified way. These results appear in chapter 8. Previously, they were presented by the author at the V Workshop School on Quantum Information and Computation (WECIQ ¹), and they are compiled in the article entitled "Classical Probabilistically Checkable Proof and Multi-Prover Quantum Merlin-Arthur" in its conference proceedings.

Controlling entanglement correlations or breaking them in a computationally efficient way may lead to the collapse $\text{QMA}(k)$ being equal to QMA. A generic quantum operation for “breaking” entanglement is known as disentangler. In a similar vein, quantum de Finetti theorems provide closeness guaranties to non-entangled states for states that obey

¹WECIQ in Portuguese.

certain properties. We rule out the existence of a range of disentanglers and de Finetti theorems under certain hardness assumptions. These results appear in chapter 9. They were also presented at the V WEICQ, as in the article "On the Hardness of Disentanglers and Quantum de Finetti Theorems", also in its conference proceedings.

Despite being an important research topic in Physics, the area law has received major contributions from Computer Science. Roughly speaking, given a composite quantum system satisfying some properties and a region dividing it into two subsystems A and \bar{A} , this law conjectures that the entanglement across the cut (A, \bar{A}) scales at most with the area of the region rather than its volume. It provides an illustration of how the properties of the underlying physical system can be used to bound entanglement. This conjecture was proved for one dimensional systems, and it is surveyed in chapter 10.

Before exposing the reader to new or more advanced results, we provide an extensive foundation of quantum computing and complexity intended to make this thesis a useful reference for a broader audience². However, this exposition is far from exhaustive and the reader may also refer to [74] [63] [98] for quantum theory background and to [10] [75] [71] for classical theory. A brief exposition of the fundamentals of quantum computing is presented in the next chapter. To motivate the study of entanglement, this property is formally defined and several basic applications are discussed in chapter 3. Quantum information requires different tools for its manipulation. For this reason, important tools are surveyed in chapter 4. Complexity classes are explored in chapter 5 bringing a pictorial view of the relationship among relevant classical and quantum classes with the placement of $\text{QMA}(k)$ among them. The classical PCP Theorem and some equivalent formulations are treated in chapter 6.

After the complexity exposition, we present some computability results. It is possible to augment a classical finite state automaton (DFA) with a finite dimensional quantum memory on which only a fixed set of unitaries can act. The Two-way Quantum Classical Finite State Automaton (2QCFA) is one such generalization. Trivially, it is at least as powerful as any classical DFA. Nevertheless, due to its quantum memory, it can also recognize some non-regular and non-context free languages. We survey recognizable languages and properties of this model in chapter 11. Moreover, we extend this model with the ability of placing a constant number of markers in the input tape. We show that for any number of markers the quantum variant is more powerful than its deterministic and probabilistic classical analogues, thus closing an open question. These new results are presented in chapter 12.

Recapitulating, our contributions are concentrated on chapters:

- Chapter 7, Variations of $\text{QMA}(k)$;

²This was one of our goals with FAPESP, our research supporting agency.

- Chapter 8, $\text{QMA}(k)$ and classical PCPs;
- Chapter 9, Hardness of disentanglers and de Finetti theorems; and
- Chapter 12, Multi-marker quantum finite automata.

The investigation of quantum finite automata was greatly benefited by the close collaboration with Prof. Moura who is the author's advisor. Whereas the investigation in Quantum Complexity gained momentum while the author was a research intern under the supervision of Prof. Thomas Vidick at the Caltech's Institute for Quantum Information and Matter (IQIM).

Chapter 2

Foundations

Quantum Computing is a multidisciplinary field at the intersection of Physics, Mathematics, and Computer Science. The latter gives a myriad of useful operational interpretations to quantum phenomena. These phenomena are governed by the postulates of Quantum Mechanics which captures the state-of-the-art understanding of the world at small scales. The quantum theory attributes physical meaning to mathematical objects from primarily linear algebra, probability theory, and differential equations. For this reason, mathematics plays a crucial role in the study of Quantum Computing. Ultimately, a quantum algorithm can be represented by a family of matrices satisfying some properties. To leverage the potential of the quantum model, a computer scientist must work on the circuit level. The layers of abstractions, well established in classical computing, do not leave behind unforeseen possibilities to enhance computing capabilities. In the quantum case, it is not clear whether abstractions would not obfuscate the potential gains of this new paradigm.

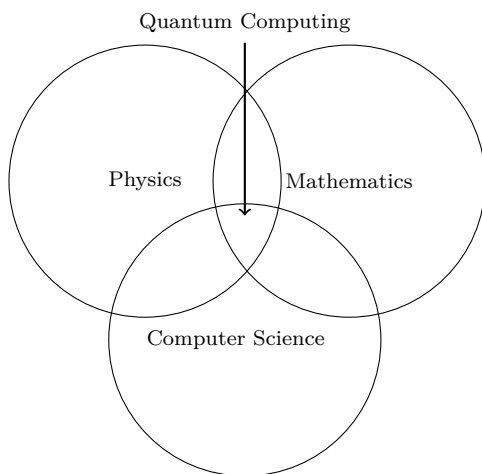


Figure 2.1: Position of Quantum Computing among sciences.

This foundation section was conceived to provide the readers a bare minimum background on quantum computing. The widely used Dirac notation for linear algebra is introduced in section 2.1. Next, the postulates of Quantum Mechanics that fix the rules of this computational paradigm are presented in section 2.2. A basic case of the Schrödinger equation for the time evolution operator illustrates the role of Hamiltonians in section 2.3. Hamiltonian Complexity is a major area of investigation in Quantum Complexity. In section 2.4, we move to the Bloch sphere that provides a pictorial way for visualizing the state of a single qubit. The quantum circuit model, which is a general purpose model for quantum computation, is presented in section 2.5. Finally, in section 2.6, we show how quantum states can be compared using measures that allow important operational interpretations.

2.1 Dirac Notation

The Dirac notation is predominant in Quantum Mechanics and Quantum Computing. It establishes a notation for vectors and dual vectors that emphasizes the inner product. The notation $|\rangle$ designates the ket whereas $\langle|$ designates the bra. Inside the delimiters of a ket and a bra, we place a label, usually Greek letters. The object $|\psi\rangle$ represents a column vector while $\langle\psi|$ represents its dual which is a line vector. Using this notation, a basis for \mathbb{C}^2 can be represented as

$$|0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}, |1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix},$$

and their duals are

$$\langle 0| = (1 \ 0), \langle 1| = (0 \ 1).$$

The inner product (bracket) of two vectors is a bra multiplied by a ket (bra-ket) of appropriate dimensions. This is the reason why the Dirac notation uses the term ket for column vectors and bra for line vectors. The inner product $\langle\phi|\psi\rangle$ is conjugate linear in the first argument ($|\phi\rangle$) and linear in the second argument ($|\psi\rangle$). For a general vector in \mathbb{C}^2 , like $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$, its dual is then $\langle\psi| = \alpha^*\langle 0| + \beta^*\langle 1|$. Computing the inner product of $|\psi\rangle$ with itself, we have $\langle\psi|\psi\rangle = |\alpha|^2 + |\beta|^2$.

The rank one projector on the space spanned by $|\psi\rangle$ is the operator $|\psi\rangle\langle\psi|$. For any orthonormal basis $\{|i\rangle\}$, the identity operator can be written as $\mathbb{I} = \sum_i |i\rangle\langle i|$.

The set of linear operators from a vector space \mathcal{M} to another vector space \mathcal{N} is denoted by $L(\mathcal{M}, \mathcal{N})$. When \mathcal{M} is the same of \mathcal{N} , we use the notation $L(\mathcal{N})$. Let $A \in L(\mathbb{C}^n)$. Fixing an orthonormal basis $\{|i\rangle\}$ for \mathbb{C}^n , we can write A in this basis as

$$A = \sum_{i,j} \langle i|A|j\rangle |i\rangle\langle j|.$$

Let $B \in L(\mathbb{C}^n)$. As an warm up to the Dirac notation, we show the cyclic property $\text{Tr}(AB) = \text{Tr}(BA)$ as

$$\begin{aligned}
\text{Tr}(AB) &= \text{Tr}\left(\left(\sum_{i,j} \langle i|A|j\rangle |i\rangle\langle j|\right) \left(\sum_{i',j'} \langle i'|B|j'\rangle |i'\rangle\langle j'|\right)\right) \\
&= \sum_k \sum_{i,j} \sum_{i',j'} \langle i|A|j\rangle \langle i'|B|j'\rangle \langle k|i\rangle \langle j|i'\rangle \langle j'|k\rangle \\
&= \sum_{i,j} \sum_{i',j'} \langle i|A|j\rangle \langle i'|B|j'\rangle \langle j|i'\rangle \langle j'| \left(\sum_k |k\rangle\langle k|\right) |i\rangle \\
&= \sum_{i,j} \sum_{i',j'} \langle i|A|j\rangle \langle i'|B|j'\rangle \langle j| \left(\sum_k |k\rangle\langle k|\right) |i'\rangle \langle j'| |i\rangle \\
&= \sum_k \sum_{i,j} \sum_{i',j'} \langle i|A|j\rangle \langle i'|B|j'\rangle \langle k|i'\rangle \langle j'| |i\rangle \langle j|k\rangle \\
&= \sum_k \langle k| \left(\sum_{i',j'} \langle i'|B|j'\rangle |i'\rangle\langle j'|\right) \left(\sum_{i,j} \langle i|A|j\rangle |i\rangle\langle j|\right) |k\rangle \\
&= \text{Tr}(BA).
\end{aligned}$$

2.2 Postulates

We establish the postulates of Quantum Mechanics tailored to our study in Quantum Computing (QC). Linear algebra and basic probability theory form the mathematical backbone of these postulates. Despite their apparently simplicity, they engender a wealthy of intricate behaviour and applications. Even for finite dimensions which is the main scenario for QC, there are a plethora of open questions. Unpacking the ideas conveyed by these simple postulates requires a great deal of study.

2.2.1 Closed System State

A Hilbert space is a vector space equipped with an inner product¹. A physical systems is said to be closed (or isolated) if it does not interact with its surrounding environment. The state of a closed quantum system lives in a complex Hilbert space as postulated next.

Postulate 1 (Closed State). *The state of a closed quantum system is given by a family of unity vectors in a complex Hilbert space. The vectors in this family are equivalent up to a global phase.*

An isolated two-level quantum system is a unity vector $|\psi\rangle$ in \mathbb{C}^2 . Using the basis $\{|0\rangle, |1\rangle\}$, we can represent it as $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ where $|\alpha|^2 + |\beta|^2 = 1$. Note that we

¹For non-finite dimensional vector spaces, it must also be complete with respect to the norm induced by the inner product.

have the equivalence $e^{i\theta}|\psi\rangle \equiv |\psi\rangle$ for $\theta \in \mathbb{R}$. In QC, a two-level quantum system is a quantum bit or simply a qubit. Contrary to its classical analogue that is either $|0\rangle$ or $|1\rangle$, a qubit can be in a coherent superposition of these two states. Note that classical bits can only be in a probabilistic superposition.

It is possible to generalize the two-level quantum system to an arbitrary number of levels d . This kind of system is called a qudit. The state of a d -level qudit can be represented as $|\psi\rangle = \sum_{i=0}^{d-1} \alpha_i |i\rangle$ where $\sum_{i=0}^{d-1} |\alpha_i|^2 = 1$.

2.2.2 Simple Measurement

It is possible to measure a quantum state in any orthonormal basis. The outcome of this process is inherently probabilistic as postulated by Quantum Mechanics. This has the philosophical implication that no simulation of the world, no matter how accurate, can predict with certainty its evolution as opposed to Laplace's ideas [98]. However, it has the bright side of ensuring the existence of true randomness.

Postulate 2 (Simple Measurement). *When a closed quantum system $|\psi\rangle = \sum_{i=0}^{d-1} \alpha_i |i\rangle$ is measured in an orthonormal $\{|k\rangle\}$, the state collapses to $|k\rangle$ with probability $|\langle k|\psi\rangle|^2$.*

Due to this postulate, it is natural to call the complex coefficients α_i of $|\psi\rangle$ the probability amplitudes. It is not a true probability, but when $|\psi\rangle$ is measured in the basis $\{|i\rangle\}$ it collapses to $|i\rangle$ with probability $|\alpha_i|^2$. A ubiquitous basis is $\{|x\rangle : x \in \{0, 1\}^n\}_n$ which is known as the computational basis of n qubits.

In physics, it is common to measure a state $|\psi\rangle$ using an observable which is simply an Hermitian operator. This kind of operator admits a spectral decomposition. Let A be an observable and $A = \sum_j \lambda_j |j\rangle\langle j|$ its spectral decomposition. The outcome of the measurement is given by the eigenvalues λ_j which happen with probability $|\langle j|\psi\rangle|^2$. It is a generalization of the simple measurement postulate. The requirement that an observable be Hermitian is placed because physical quantities such as momentum and position are usually assumed to be real and Hermitian operators have a real spectrum.

Claim 2.2.1. *Hermitian operators have a real spectrum.*

Proof. Let $|i\rangle$ be an eigenvector of A with corresponding eigenvalue λ_i . We have

$$\lambda_i = \langle i|A|i\rangle = \langle i|A^\dagger|i\rangle = \lambda_i^*.$$

□

The expected value of an observable $A = \sum_j \lambda_j |j\rangle\langle j|$ given a state $|\psi\rangle$ is

$$\langle A \rangle_{|\psi\rangle} = \sum_j \lambda_j |\langle j|\psi\rangle|^2.$$

2.2.3 Closed System Evolution

Due to the closed state postulate 1, it is natural to enforce that the evolution of a closed system must preserve its norm.

Postulate 3 (Closed Evolution). *The state of a closed quantum system evolves through the application of unitary operations.*

Let $|\psi\rangle$ be a quantum state evolving through the unitary U . The following invariant holds

$$\langle\psi|\psi\rangle = \langle\psi|U^\dagger U|\psi\rangle.$$

Therefore, a unitary operator must satisfy $U^\dagger U = \mathbb{I}$. Let $\{|i\rangle\}_{i=1}^n$ and $\{|i'\rangle\}_{i=1}^n$ be orthonormal basis. Let $\sigma : [n] \rightarrow [n]$ be a permutation. The operator $U = \sum_{i=1}^n |\sigma(i)'\rangle\langle i|$ satisfy

$$U^\dagger U = \sum_{i,j} |i\rangle\langle\sigma(i)'||\sigma(j)'\rangle\langle j| = \sum_i |i\rangle\langle i| = \mathbb{I},$$

implying that it is unitary. Rotations and reflections are also examples of unitary operators. Note that unitary operators admit a spectral decomposition in which the eigenvalues are complex values of norm one. Another important property of unitary evolution is that it is reversible as unitaries always have an inverse which is also unitary.

2.2.4 Composite System State

One benefit of working with the quantum model appears when quantum systems are combined. If we combine n individual qubits, their final state live in a Hilbert space of dimension 2^n as captured by the following postulate.

Postulate 4 (Composite State). *When two systems that live in spaces \mathcal{H}_1 and \mathcal{H}_2 are combined, the final state lives in space $\mathcal{H}_1 \otimes \mathcal{H}_2$.*

The tensor product of two qubits $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ and $|\phi\rangle = \alpha'|0\rangle + \beta'|1\rangle$ is

$$\begin{aligned} |\psi\rangle \otimes |\phi\rangle &= \alpha\alpha'|0\rangle \otimes |0\rangle + \alpha\beta'|0\rangle \otimes |1\rangle + \beta\alpha'|1\rangle \otimes |0\rangle + \beta\beta'|1\rangle \otimes |1\rangle \\ &= \alpha\alpha'|00\rangle + \alpha\beta'|01\rangle + \beta\alpha'|10\rangle + \beta\beta'|11\rangle, \end{aligned}$$

where $|i\rangle \otimes |j\rangle = |ij\rangle$.

Let $\mathcal{H}_1 = \text{span}(B_1)$ and $\mathcal{H}_2 = \text{span}(B_2)$ where $B_1 = \{|i\rangle\}$ and $B_2 = \{|j\rangle\}$ are orthonormal basis for \mathcal{H}_1 and \mathcal{H}_2 . The space $\mathcal{H}_1 \otimes \mathcal{H}_2 = \text{span}(B_3)$ where $B_3 = \{|i\rangle \otimes |j\rangle : |i\rangle \in B_1, |j\rangle \in B_2\}$. If \mathcal{H}_1 and \mathcal{H}_2 are the spaces of two individual qubits, then the final

space has dimension 4. Adding an extra qubit doubles the dimensional of the resulting space.

The tensor product also applies to matrices. Let A be

$$A = \begin{pmatrix} a_{1,1} & \dots & a_{1,n} \\ \vdots & \ddots & \vdots \\ a_{m,1} & \dots & a_{m,n} \end{pmatrix},$$

the tensor product of A and another matrix B is

$$A \otimes B = \begin{pmatrix} a_{1,1}B & \dots & a_{1,n}B \\ \vdots & \ddots & \vdots \\ a_{m,1}B & \dots & a_{m,n}B \end{pmatrix}.$$

2.2.5 Open System State

The closed system state postulate guarantees the existence of the coherent superposition $|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$. Intuitively, it seems physically conceivable to create a probabilistic state by outputting with equal probability $|0\rangle$ and $|1\rangle$. This postulate does not capture this kind of probabilistic state implying that they are not from a closed system. Nevertheless, it is important to have a more general representation of quantum states.

Postulate 5 (General State). *A quantum state can be represented by a density operator which is a positive semidefinite operator of unity trace.*

We denote the set of density operators in $\mathbb{C}^n \times \mathbb{C}^n$ by $D(\mathbb{C}^n)$. Since the operators we cover in this work are linear, we use the terms operator and matrix interchangeably.

A pure state $|\psi\rangle$ corresponds to the density operator $|\psi\rangle\langle\psi|$. The states $|0\rangle$ and $|1\rangle$ correspond to $|0\rangle\langle 0|$ and $|1\rangle\langle 1|$, respectively. The probabilistic procedure that generates $|0\rangle\langle 0|$ and $|1\rangle\langle 1|$ with equal probability results in the density operator $\rho = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1|$. On the other hand, the state $|\psi\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$ becomes the density operator $\psi = \frac{1}{2}|0\rangle\langle 0| + \frac{1}{2}|0\rangle\langle 1| + \frac{1}{2}|1\rangle\langle 0| + \frac{1}{2}|1\rangle\langle 1|$. For a pure state $|\xi\rangle$, be aware that it is common to use ξ to denote its density operator.

As expected, any convex combination of density operators is also a valid density operator. Let p be a probability distribution over Γ and $\{\sigma_x : x \in \Gamma\}$ be a set of density operators. The convex combination has trace

$$\text{Tr}\left(\sum_{x \in \Gamma} p(x)\sigma_x\right) = \sum_{x \in \Gamma} p(x) \text{Tr}(\sigma_x) = \sum_{x \in \Gamma} p(x) = 1.$$

Moreover, for any $|\psi\rangle$ we have

$$\langle\psi| \sum_{x \in \Gamma} p(x)\sigma_x |\psi\rangle = \sum_{x \in \Gamma} p(x) \langle\psi|\sigma_x|\psi\rangle \geq 0,$$

proving that the convex combination is also positive semidefinite.

2.2.6 General Measurement

Despite not being a postulate, it is useful to define the most general form of quantum measurement known as POVM.

Definition 2.2.2 (POVM). *A quantum state can be measured using a set of operators $\{M_x\}$ denoted Positive Operator Value Measurement (POVM). These operators must satisfy two conditions:*

$$\sum_x M_x = \mathbb{I},$$

and

$$0 \leq M_x \leq \mathbb{I}, \text{ for all } x.$$

The index x of M_x is said to be the outcome of the measurement. For a state ρ , the probability of measuring x is

$$\text{Tr}(M_x \rho).$$

In which case, the state collapses to

$$\frac{\sqrt{M_x} \rho \sqrt{M_x}}{\text{Tr}(M_x \rho)}.$$

2.2.7 Subsystem State

As important as understanding what happens to a quantum state when two systems A and B are combined, it is to understand when one of them is removed. In the quantum jargon, the action of removing a system is known as tracing out. The next postulate establishes how to obtain the traced-out state.

Postulate 6. *Let ρ^{AB} be density operators on systems A and B . The state of system A is*

$$\rho^A = \text{Tr}_B(\rho^{AB}),$$

where $\text{Tr}_B(\rho^{AB})$ is the partial trace of ρ^{AB} over system B .

The partial trace is a linear operation satisfying

$$\text{Tr}_B(|i\rangle\langle j|^A \otimes |k\rangle\langle l|^B) = |i\rangle\langle j|^A \text{Tr}(|k\rangle\langle l|^B) = |i\rangle\langle j|^A \langle l|k\rangle.$$

Due to linearity it is straightforward to extend the previous definition to an arbitrary operator ρ^{AB} . Let $\{|i\rangle^A\}$ and $\{|j\rangle^B\}$ be basis for systems A and B , respectively. We can write ρ^{AB} as

$$\rho^{AB} = \sum_{i,i',j,j'} (\langle i|\langle j|\rho^{AB}|i'\rangle|j'\rangle) |i\rangle\langle i'|^A \otimes |j\rangle\langle j'|^B.$$

Tracing out system B , we obtain state ρ^A as

$$\begin{aligned} \rho^A &= \text{Tr}_B(\rho^{AB}) = \sum_{i,i',j,j'} (\langle i|\langle j|\rho^{AB}|i'\rangle|j'\rangle) |i\rangle\langle i'|^A \otimes \text{Tr}(|j\rangle\langle j'|^B) \\ &= \sum_{i,i',j} (\langle i|\langle j|\rho^{AB}|i'\rangle|j\rangle) |i\rangle\langle i'|^A. \end{aligned}$$

Note that $\text{Tr}(\rho^A)$ is one as expected

$$\begin{aligned} \text{Tr}(\rho^A) &= \sum_{i,i',j,k} (\langle i|\langle j|\rho^{AB}|i'\rangle|j\rangle) \langle k|i\rangle\langle i'|k\rangle \\ &= \sum_{i,i',j} (\langle i|\langle j|\rho^{AB}|i'\rangle|j\rangle) \langle i'|(\sum_k |k\rangle\langle k|)|i\rangle \\ &= \sum_{i,j} \langle i|\langle j|\rho^{AB}|i\rangle|j\rangle \\ &= \text{Tr}(\rho^{AB}) = 1. \end{aligned}$$

The state ρ^A is also positive semidefinite

$$\begin{aligned} \langle\psi|\rho^A|\psi\rangle &= \sum_{i,i',j} (\langle i|\langle j|\rho^{AB}|i'\rangle|j\rangle) \langle\psi|i\rangle\langle i'|\psi\rangle \\ &= \sum_j \langle\psi|(\sum_i |i\rangle\langle i|)\langle j|\rho^{AB}(\sum_{i'} |i'\rangle\langle i'|)|\psi\rangle|j\rangle \\ &= \sum_j \langle\psi|\langle j|\rho^{AB}|\psi\rangle|j\rangle \geq 0, \end{aligned}$$

since ρ^{AB} is positive semidefinite by hypothesis. This implies that tracing out quantum density operators results in valid quantum states.

We analyze the partial trace for a more restricted type of density operator that is recurrent in the study of quantum computing. If $\rho^{AB} = \rho^A \otimes \rho^B$, it follows that

$$\text{Tr}_B(\rho^{AB}) = \rho^A \otimes \text{Tr}(\rho^B) = \rho^A.$$

So far, we have only traced out system B . Observe that tracing out system A is completely analogous.

Suppose we have at our disposal the state ρ^{AB} but we want to measure only the system A using the POVM $\{M_x^A\}$. This is equivalent to measure ρ^{AB} with $\{M_x^A \otimes \mathbb{I}^B\}$. It is expected that measuring ρ^A with $\{M_x^A\}$ would result in the same measuring statistics. The probability of outcome x is

$$\begin{aligned}
p(x) &= \text{Tr}((M_x^A \otimes \mathbb{I}^B)\rho^{AB}) \\
&= \text{Tr}((M_x^A \otimes \mathbb{I}^B)(\sum_{i,i',j,j'} (\langle i|\langle j|\rho^{AB}|i'\rangle|j'\rangle)|i\rangle\langle i'|^A \otimes |j\rangle\langle j'|^B)) \\
&= \sum_{i,i',j,j'} \langle i|\langle j|\rho^{AB}|i'\rangle|j'\rangle \text{Tr}(M_x^A|i\rangle\langle i'|^A \otimes |j\rangle\langle j'|^B) \\
&= \sum_{i,i',j,j'} \langle i|\langle j|\rho^{AB}|i'\rangle|j'\rangle \text{Tr}(M_x^A|i\rangle\langle i'|^A) \text{Tr}(|j\rangle\langle j'|^B) \\
&= \sum_{i,i',j} \langle i|\langle j|\rho^{AB}|i'\rangle|j\rangle \text{Tr}(M_x^A|i\rangle\langle i'|^A) = \text{Tr}(M_x^A \rho^A).
\end{aligned}$$

2.2.8 Open System Evolution

A physical system is said to be open when it is allowed to interact with the surrounding environment. The evolution of an open quantum system is the most general form of quantum evolution. It is governed by a linear operator that maps the set of quantum density operators to itself. Moreover, when applied to a subsystem of a larger system, the combined final state must remain positive. If an operator satisfy this last property, we say that it is completely positive. The next postulate formalizes the notion of a quantum superoperator.

Postulate 7 (Open System Evolution). *The evolution of an open system is given by a linear operator which is completely positive and trace preserving.*

The particular case in which the state ρ evolves through a unitary operation U can be expressed as

$$\Phi(\rho) = U\rho U^\dagger.$$

Every valid quantum superoperator $\Phi : D(\mathbb{C}^n) \rightarrow D(\mathbb{C}^m)$ can be specified by a set a $\{E_k\}_{k=1}^{nm}$ where each $E_k \in L(\mathbb{C}^n, \mathbb{C}^m)$ and $\sum_k E_k^\dagger E_k = \mathbb{I}$. This specification is known as the Krauss representation. Using it, the mapping Φ becomes

$$\Phi(\rho) = \sum_{k=1}^{nm} E_k \rho E_k^\dagger.$$

It is straightforward to verify that Φ is trace preserving as

$$\text{Tr}(\Phi(\rho)) = \sum_{k=1}^{nm} \text{Tr}(E_k \rho E_k^\dagger) = \sum_{k=1}^{nm} \text{Tr}(E_k^\dagger E_k \rho) = \text{Tr}(\rho) = 1.$$

Instead of the Krauss representation, it is possible to assume, without loss of generality, that every evolution of a state $\rho \in \mathcal{D}(\mathbb{C}^n)$ is governed by a unitary [74]. However, this unitary does not act only on ρ . It also affects an environment state that lives in $\mathcal{D}(\mathbb{C}^{2n})$. After the application of this unitary, all subsystems are discarded but system B which contains the output state of the given superoperator. In mathematical notation, this is equivalent to

$$\Phi(\rho) = \text{Tr}_{\setminus B}(U^{AE}(\rho^A \otimes \sigma^E)(U^{AE})^\dagger),$$

where σ^E is the environment state.

2.3 Schrödinger Equation

The Schrödinger equation for the temporal evolution operator is a differential equation of central importance in Quantum Physics. One of its components is an operator known as a Hamiltonian which also plays a crucial role in Quantum Complexity. Remember from the postulates that the evolution of a closed quantum system is governed by a unitary operator. Using this and other properties, we derive this Schrödinger equation in a similar fashion as done in [88].

Let $U(t, t_0)$ denote the unitary operator responsible for the temporal evolution from time t_0 to t . For $t = t_0 + dt$, if dt goes to zero, we would expect this operator to converge to the identity, that is

$$\lim_{dt \rightarrow 0} U(t_0 + dt, t_0) = \mathbb{I}. \quad (2.1)$$

The evolution from t_0 to t_2 must be equal to the composition of the evolutions from t_0 to t_1 and from t_1 to t_2 for $t_0 \leq t_1 \leq t_2$. More formally, $U(t_2, t_0)$ must satisfy

$$U(t_2, t_0) = U(t_2, t_1)U(t_1, t_0). \quad (2.2)$$

Defining $U(t + dt, t)$ as

$$U(t + dt, t) = \mathbb{I} - i\Omega dt,$$

the limit in Eq. 2.1 follows. Moreover, if Ω is Hermitian the unitarity also holds since

$$\begin{aligned} U(t + dt, t)^\dagger U(t + dt, t) &= (\mathbb{I} - i\Omega dt)^\dagger (\mathbb{I} - i\Omega dt) \\ &= \mathbb{I} + i\Omega^\dagger dt - i\Omega dt + \Omega^\dagger \Omega dt^2 \\ &\approx \mathbb{I}, \end{aligned}$$

where terms on dt^2 are considered small enough to be discarded. To be consistent with classical equations we need to set $\Omega = \frac{H}{\hbar}$ where H is also an Hermitian operator known as a Hamiltonian [88].

Using the composition property in Eq. 2.2 and setting $t_2 = t + dt$ and $t_1 = t$, we have

$$\begin{aligned} U(t + dt, t_0) &= U(t + dt, t)U(t, t_0) \\ &= (\mathbb{I} - i\frac{H}{\hbar}dt)U(t, t_0). \end{aligned}$$

Rearranging, give us

$$\frac{U(t + dt, t_0) - U(t, t_0)}{dt} = -i\frac{H}{\hbar}U(t, t_0).$$

In the limit $dt \rightarrow 0$, the left hand side becomes a partial derivative, and we finally recover the Schrödinger equation for the temporal operator

$$i\hbar \frac{\partial U(t, t_0)}{\partial t} = HU(t, t_0).$$

If the Hamiltonian is constant, this equation admits the simple solution

$$U(t, t_0) = e^{-i\frac{H}{\hbar}(t-t_0)},$$

which is a unitary operator with the same support of H .

2.4 Bloch Sphere

One qubit quantum states can be represented in the real Euclidean space \mathbb{R}^3 using the Bloch sphere. This representation can be used for pure and mixed states. For pure states we work on the surface of a unity ball while for mixed states we also need its interior making the term sphere a misnomer.

First, we address the pure state case in which the qubit state lives in \mathbb{C}^2 . Let $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$ be this state. Since $|\alpha|^2 + |\beta|^2 = 1$, we can write

$$\begin{aligned} \alpha &= e^{i\phi_1} \cos(\theta), \text{ and} \\ \beta &= e^{i\phi_2} \sin(\theta), \end{aligned}$$

for angles ϕ_1 , ϕ_2 , and θ . With these coefficients the pure state becomes $|\psi\rangle = e^{i\phi_1}(\cos(\theta)|0\rangle + e^{i(\phi_2-\phi_1)}\sin(\theta)|1\rangle)$, or simply $|\psi\rangle = \cos(\theta)|0\rangle + e^{i\phi}\sin(\theta)|1\rangle$ because quantum states are equivalent up to a global phase. In order to determine the intervals of θ and ϕ we will

use this equivalence again. In principle θ can range from 0 to 2π . However, note that the equivalences hold

$$\begin{aligned}\cos(\theta)|0\rangle + e^{i\phi}\sin(\theta)|1\rangle &\equiv -\cos(\theta)|0\rangle - e^{i\phi}\sin(\theta)|1\rangle, \text{ and} \\ \cos(\theta)|0\rangle - e^{i\phi}\sin(\theta)|1\rangle &\equiv -\cos(\theta)|0\rangle + e^{i\phi}\sin(\theta)|1\rangle.\end{aligned}$$

It is enough to let θ range from 0 to $\frac{\pi}{2}$, and then use ϕ to control the relative phase. Denote by ϕ' be original phase angle. We take ϕ such that $e^{i\phi} = e^{i(\phi'+\pi)}$ and $\phi \in [0, 2\pi]$ resulting in

$$\cos(\theta)|0\rangle + e^{i\phi'}e^{i\pi}\sin(\theta)|1\rangle = \cos(\theta)|0\rangle - e^{i\phi'}\sin(\theta)|1\rangle.$$

To map $|\psi\rangle$ to a real three dimensional sphere, we need one additional modification. We replace θ by $\frac{\theta}{2}$ as

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle,$$

allowing this angle to range from 0 to π . Now, except by $|0\rangle$ and $|1\rangle$, the parameters θ and ϕ unequivocally determine a point on the \mathbb{R}^3 unit sphere. Let \hat{x} , \hat{y} , and \hat{z} be orthonormal axis in \mathbb{R}^3 satisfying $\hat{x} \times \hat{y} = \hat{z}$. Then, θ is the angle between $|\psi\rangle$ and $|z\rangle$ while ϕ is the angle between the projection of $|\psi\rangle$ on the plane spanned by $\{\hat{x}, \hat{y}\}$ and the vector \hat{x} as shown in Figure 2.2.

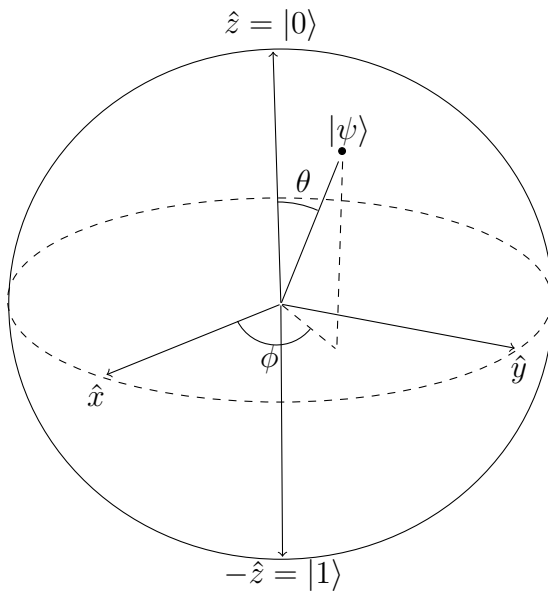


Figure 2.2: The representation of a vector $|\psi\rangle$ in the Bloch Sphere.

The state of a mixed qubit is a positive semidefinite operator of trace one in $\mathbb{C}^2 \times \mathbb{C}^2$. It can be represented by the matrix

$$\rho = \frac{1}{2} \begin{pmatrix} 1+c & a-bi \\ a+bi & 1-c \end{pmatrix},$$

in which a , b and c are reals. To give an alternative representation, we introduce the Pauli matrices

$$X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, Y = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}.$$

Using these matrices, any qubit density operator can be written as

$$\rho = \frac{1}{2}(\mathbb{I} + aX + bY + cZ)$$

Note that the trace of ρ is one. We compute its eigenvalues using the characteristic equation as

$$\det |\rho - \lambda \mathbb{I}| = \frac{-a^2 - b^2 + 1 - c^2}{4} - \frac{(1+c)}{2}\lambda - \frac{(1-c)}{2}\lambda + \lambda^2 = 0.$$

Simplifying, we have

$$\lambda^2 - \lambda + \frac{1 - |\vec{p}|^2}{4} = 0,$$

where $\vec{p} = (a, b, c)$. Thus, its zeros are

$$\lambda = \frac{1 \pm |\vec{p}|}{2}.$$

To ρ be a positive semidefinite matrix, $|\vec{p}|$ must be at most one. It means that \vec{p} lies in a \mathbb{R}^3 unity ball. If $|\vec{p}| = 1$, the two eigenvalues are 1 and 0, and thus the state is pure. In this case, its representation actually lies on the sphere. Otherwise, we have two non-zero eigenvalues and the state lies strictly in the interior of the unity ball. This is the reason why the term Bloch Sphere is a misnomer for the general case. Observe that the qubit state which is the farthest from being pure, *i.e.* $\frac{1}{2}\mathbb{I}$, is mapped to the origin.

2.5 Circuits

Classical computation is typically defined in terms of Turing Machines. Despite being possible to define quantum computation in terms of quantum Turing Machines [27], it is usually defined in terms of circuits. These two quantum models are equivalent in

expressive power [101], but the latter is considered simpler. For this reason, the quantum circuit model is the prevalent one.

The evolution of closed quantum systems is governed by unitary operators. To make the model realistic it is necessary to assume that we only dispose of a fixed set of unitaries also known as a gate set. This is analogous to classical computation, where each function can be implemented using the gate sets $\{AND, OR, NOT\}$, $\{NAND\}$, or $\{NOR\}$. Somehow, any unitary needs to be approximated by combining unitaries from a fixed gate set. If a gate set can approximate any given unitary, it is called a universal gate set. The quantum gate set $\{H, T, CNOT\}$ is one such example. We will define each of its gates next. One important remark is that quantum circuits are reversible due to unitarity. This is in sharp contrast with the classical gates AND and OR that are not reversible, *i.e.* it is not possible to always uniquely determine their inputs given the output.

The matrix representations of quantum circuits will be presented in the computational basis. The Hadamard gate, or simply H , is the single qubit gate

$$H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}.$$

When applied to $|0\rangle$, it results in $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$. This state is also denoted by $|+\rangle$. Similarly, $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$, equivalently denoted by $|-\rangle$. Since the Hadamard gate can create uniform superpositions, it can, in principle, be used to generate perfect random bits.

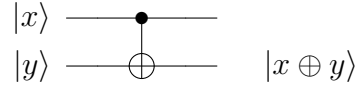
It is possible to apply a phase shift of $e^{i\frac{\pi}{4}}$ to $|1\rangle$ while letting $|0\rangle$ invariant. The gate that produces this transformation is denoted by T and it has the following matrix representation:

$$T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\frac{\pi}{4}} \end{pmatrix}.$$

We have shown so far only one-qubit gates. To be a universal gate set, it needs at least one gate acting on two or more qubits as this kind of gate can create correlations among the probability amplitudes of different qubits. The controlled not gate, or simply $CNOT$ gate, is the final gate in our universal set. It receives as input two qubits x and y where the first is the control bit and the second is the target. If x is $|1\rangle$, the gate applies the not operation to the target y . Otherwise, the identity operator is applied to qubit y . In matrix form, we have

$$CNOT = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}.$$

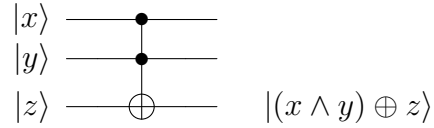
The *CNOT* circuit is depicted as.



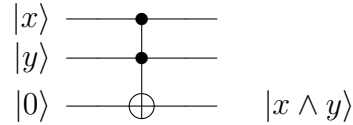
The Toffoli gate is not part of our universal set, but it illustrates how classical computation can be performed in a reversible way using the quantum circuit model. It is an extension of the *CNOT* gate in which one extra control qubit is added. Qubits x and y are now control qubits while z is the target qubit. If $x = y = |1\rangle$, a not is applied to z . Otherwise, the identity operator is applied to z . The Toffoli matrix resembles the *CNOT* one

$$\text{Toffoli} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}.$$

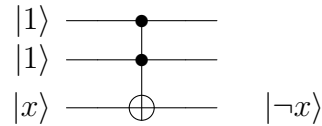
The Toffoli gate as a quantum circuit is shown next.



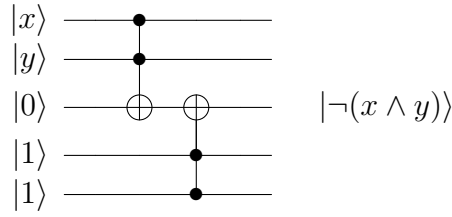
Using this gate, we show how to create reversible classical *AND* and *NOT* gates. Combining both, we get a *NAND* gate which is universal for classical computation. The circuit to compute the *AND* gate is illustrated next.



The *NOT* circuit is depicted next.



Combining both circuits, we have a reversible classical *NAND* gate.



This simulation of classical computation by the quantum model is formalized in the next claim. This means that quantum computing is at least as expressive as classical computing.

Claim 2.5.1. *If the Toffoli gate is part of the gate set in the circuit model, then any language L that can be decided by a classical Turing Machine can also be decided by a quantum machine with certainty.*

2.6 Comparing Quantum States

Quantum states are positive semidefinite operators of unity trace. For this reason, a variety of measures from mathematics can be used to compare them. Nonetheless, we focus on measures that also have important operational interpretations in quantum computing. Firstly, we present the quantum trace distance which is a generalization of the classical trace distance for distributions. Secondly, we define the fidelity which is a similarity measure also inspired in the classical fidelity for distributions. Having a good handling on these measures is paramount to understanding of the Quantum Complexity literature. In quantum proof verification, a cheating prover may provide the verifier an arbitrary quantum certificate. In this context, how can the verifier enforce that accepted certificates are close to what is ideally expected? What are the consequences of a slight different certificate in the verifier's accepting probability? These fundamental questions are closely connected to these measures.

2.6.1 Distance Measures

To better understand the trace distance, it is useful to introduce the Schatten p -norm and the operator p -norm.

Schatten p -norm

The Schatten p -norm is a generalization of the l_2 norm used in the Euclidean distance. It can be defined as follows.

Definition 2.6.1 (Schatten p -norm). *The Schatten p -norm of a vector $v \in \mathbb{C}^n$ is*

$$\|v\|_p = \left(\sum_{i=1}^n |v[i]|^p \right)^{\frac{1}{p}}.$$

Operator p -norm

The operator p -norm of a normal operator X is just the Schatten p -norm applied to its eigenvalues.

Definition 2.6.2 (Operator p -norm). *The p -norm of a normal operator X is*

$$\|X\|_p = \|\vec{\lambda}\|_p,$$

where $\vec{\lambda}$ is the vector containing the eigenvalues of X and $\|\vec{\lambda}\|_p$ is its Schatten p -norm.

Trace Norm

The trace norm is a particular case of the operator p -norm in which p is 1. The unity ball in the trace norm is contained in the unity ball of any p -norm for $p > 1$. This is a simple consequence of using Schatten p -norm for the eigenvalues.

Definition 2.6.3 (Trace Norm). *The Trace Norm of a normal operator X is its 1-norm, i.e. $\|X\|_1$.*

Trace Distance

The trace distance for classical distributions p and q is the optimum bias in distinguishing them when we are given a sample of p or q with equal probability. It is defined next.

Definition 2.6.4 (Trace Distance for Classical Distributions). *Let p and q be two distributions with labels in Γ . The trace distance of p and q is*

$$D(p, q) = \frac{1}{2} \|p - q\|_1 = \frac{1}{2} \sum_{x \in \Gamma} |p(x) - q(x)|.$$

Generalizing the previous definition, we have the trace distance for density operators which also uses the Schatten 1-norm.

Definition 2.6.5 (Trace Distance for Mixed States). *Let ρ and σ be two mixed states. The trace distance of ρ and σ is*

$$D(\rho, \sigma) = \frac{1}{2} \|\rho - \sigma\|_1 = \frac{1}{2} \|\vec{\lambda}\|_1,$$

where $\vec{\lambda}$ is the vector containing the eigenvalues of $\rho - \sigma$.

Pure quantum states admit another expression for the trace distance as shown in the next claim.

Claim 2.6.6 (Trace Distance for Pure States). *Let $|\psi\rangle$ and $|\phi\rangle$ be two pure states. The trace distance between them is*

$$D(|\psi\rangle, |\phi\rangle) = \sqrt{1 - |\langle\psi|\phi\rangle|^2}.$$

Proof. Let $|\psi\rangle = \alpha|\phi\rangle + \beta|\phi^\perp\rangle$ where $|\alpha|^2 + |\beta|^2 = 1$ and $\langle\phi|\phi^\perp\rangle = 0$. The density matrix for $|\psi\rangle$ is

$$\psi = |\alpha|^2|\phi\rangle\langle\phi| + \alpha^*\beta|\phi^\perp\rangle\langle\phi| + \alpha\beta^*|\phi\rangle\langle\phi^\perp| + |\beta|^2|\phi^\perp\rangle\langle\phi^\perp|.$$

Computing the difference of the density operators ϕ and ψ , we have

$$\phi - \psi = \begin{pmatrix} 1 - |\alpha|^2 & -\alpha\beta^* \\ -\alpha^*\beta & -|\beta|^2 \end{pmatrix} = \begin{pmatrix} |\beta|^2 & -\alpha\beta^* \\ -\alpha^*\beta & -|\beta|^2 \end{pmatrix}.$$

We compute the eigenvalues of the previous operator using its characteristic polynomial as

$$\det |(\phi - \psi) - \lambda I| = (|\beta|^2 - \lambda)(-|\beta|^2 - \lambda) - |\alpha|^2|\beta|^2 = 0.$$

This results in the eigenvalues

$$\begin{aligned} \lambda &= \pm\sqrt{|\beta|^4 + |\alpha|^2|\beta|^2} \\ &= \pm\sqrt{(1 - |\alpha|^2)|\beta|^2 + |\alpha|^2|\beta|^2} \\ &= \pm\sqrt{|\beta|^2} = \pm\sqrt{1 - |\alpha|^2} = \pm\sqrt{1 - |\langle\psi|\phi\rangle|^2}, \end{aligned}$$

thus proving the result. \square

Trace Distance Properties

The trace distance for quantum states has a very important property similar to the classical case. It gives the optimal bias in distinguishing state ρ and σ when we are given one of them with equal probability.

Lemma 2.6.7 (Trace Distance Operational Interpretation).

$$D(\rho, \sigma) = \max_{0 \leq M \leq \mathbb{I}} \text{Tr}(M(\rho - \sigma))$$

Proof. Let $\xi = \rho - \sigma$. Note that it is an Hermitian operator. Let $\sum_i \lambda_i |i\rangle\langle i|$ be its spectral decomposition. Let $P = \{i : \lambda_i > 0\}$ and $N = \{i : \lambda_i \leq 0\}$. We can rewrite ξ as

$$\xi = \sum_{i \in P} \lambda_i |i\rangle\langle i| - \sum_{i \in N} \lambda_i |i\rangle\langle i|.$$

Computing the trace of ξ , results in

$$\text{Tr}(\xi) = \text{Tr}(\rho - \sigma) = \text{Tr}(\rho) - \text{Tr}(\sigma) = 0.$$

Alternatively, we can compute this trace as

$$\begin{aligned} \text{Tr}(\xi) &= \sum_{i \in P} \lambda_i \text{Tr}(|i\rangle\langle i|) - \sum_{i \in N} \lambda_i \text{Tr}(|i\rangle\langle i|) \\ &= \sum_{i \in P} \lambda_i - \sum_{i \in N} \lambda_i. \end{aligned}$$

We conclude that $\sum_{i \in P} \lambda_i = \sum_{i \in N} \lambda_i$. For $D(\rho, \sigma)$, we have

$$D(\rho, \sigma) = \frac{1}{2} \|\xi\|_1 = \frac{1}{2} \left(\sum_{i \in P} \lambda_i + \sum_{i \in N} \lambda_i \right).$$

The optimum measurement $M = \sum_{i \in P} |i\rangle\langle i|$ gives

$$\text{Tr}(M(\rho - \sigma)) = D(\rho, \sigma).$$

□

If we want to distinguish two density operators ρ and σ which are given with equal probability, there is no unitary which would increase the probability of distinguishing them. Stated more formally, we have the following claim.

Claim 2.6.8. *Let ρ and σ be two density operators. We have*

$$D(\rho, \sigma) = D(U\rho U^\dagger, U\sigma U^\dagger)$$

for any choice of unitary U .

Proof. Let $\xi = \rho - \sigma$. The operator ξ is Hermitian, and so it admits a spectral decomposition $\xi = V\Sigma V^\dagger$ where V is unitary and Σ is diagonal. We compute $D(U\rho U^\dagger, U\sigma U^\dagger)$ as

$$\begin{aligned} D(U\rho U^\dagger, U\sigma U^\dagger) &= \|U\rho U^\dagger - U\sigma U^\dagger\|_1 \\ &= \|U(\rho - \sigma)U^\dagger\|_1 = \|U\xi U^\dagger\|_1 \\ &= \|UV\Sigma V^\dagger U^\dagger\|_1 = \sum_i |\langle i|\Sigma|i\rangle| \\ &= \|V\Sigma V^\dagger\|_1 = D(\rho, \sigma). \end{aligned}$$

□

The trace distance is convex in the first and the second argument.

Claim 2.6.9 (Convexity). *Let ρ , σ , and ξ be density operators. For $\alpha \in [0, 1]$, it holds*

$$D(\alpha\rho + (1 - \alpha)\sigma, \xi) \leq \alpha D(\rho, \xi) + (1 - \alpha)D(\sigma, \xi).$$

Proof.

$$\begin{aligned} D(\alpha\rho, (1 - \alpha)\sigma) &= \frac{1}{2} \|\alpha\rho + (1 - \alpha)\sigma - \xi\|_1 \\ &= \frac{1}{2} \|\alpha\rho + (1 - \alpha)\sigma - (\alpha\xi + (1 - \alpha)\xi)\|_1 \\ &\leq \frac{1}{2} (\alpha \|\rho - \xi\|_1 + (1 - \alpha) \|\sigma - \xi\|_1) \\ &= \alpha D(\rho, \xi) + (1 - \alpha)D(\sigma, \xi). \end{aligned}$$

□

In quantum computational complexity, a proof verifier can be specified by a binary POVM $\{M_0, M_1 = \mathbb{I} - M_0\}$. If this verifier accepts a proof ρ with probability p , it is expected that it would accept another proof σ which is close to ρ with a similar probability. This is actually the case when closeness is measured according to the trace distance.

Lemma 2.6.10. *Let $0 \leq M \leq \mathbb{I}$ be a measurement operator. Suppose ρ and σ are states such that $D(\rho, \sigma) \leq \epsilon$. If $\text{Tr}(M\rho) = p$, then $p - \epsilon \leq \text{Tr}(M\sigma) \leq p + \epsilon$.*

Proof. Let M' be a measurement such that $\text{Tr}(M'(\rho - \sigma)) = D(\rho, \sigma)$. By Lemma 2.6.7, the following inequality holds

$$\epsilon \geq D(\rho, \sigma) = \max_{0 \leq M \leq \mathbb{I}} \text{Tr}(M'(\rho - \sigma)) \geq \text{Tr}(M\rho) - \text{Tr}(M\sigma),$$

and we can conclude that $\text{Tr}(M\sigma) \geq p - \epsilon$. The case $\text{Tr}(M\sigma) \leq p + \epsilon$ is proved in an analogous way. □

Suppose we want to distinguish two photos of the same size. Having half of each photo is no better than having the complete photos for this task. Similarly, the trace distance of two quantum operators is no smaller than the trace distance of their reduced density operators. For this reason, we say that this distance is contractible.

Claim 2.6.11. *Let ρ^{AB} and σ^{AB} be two density operators on systems A and B . The trace distance is a contractible operation that is*

$$D(\rho^{AB}, \sigma^{AB}) \geq D(\rho^A, \sigma^A),$$

where $\rho^A = \text{Tr}_B(\rho^{AB})$ and $\sigma^A = \text{Tr}_B(\sigma^{AB})$.

Proof. We use the operational property of the trace distance as

$$\begin{aligned} D(\rho^{AB}, \sigma^{AB}) &= \text{Tr}(M^{AB}(\rho^{AB} - \sigma^{AB})) \\ &\geq \text{Tr}((M'^A \otimes I^B)(\rho^{AB} - \sigma^{AB})) \\ &= \text{Tr}(M'^A(\rho^A - \sigma^A)) = D(\rho^A, \sigma^A). \end{aligned}$$

□

Let $\rho^A \otimes \rho^B$ and $\sigma^A \otimes \sigma^B$ be states which are product across systems A and B . Their distance is no greater than the distance on the individual systems A and B .

Claim 2.6.12. *Let ρ^A , σ^A , ρ^B , and σ^B be density operators. The following inequality holds*

$$D(\rho^A \otimes \rho^B, \sigma^A \otimes \sigma^B) \leq D(\rho^A, \sigma^A) + D(\rho^B, \sigma^B).$$

Proof. First, we show that $D(\rho^A \otimes \xi^B, \sigma^A \otimes \xi^B) = D(\rho^A, \sigma^A)$ by computing

$$\begin{aligned} D(\rho^A \otimes \xi^B, \sigma^A \otimes \xi^B) &= \max_{0 \leq M^{AB} \leq \mathbb{I}} \text{Tr}(M^{AB}(\rho^A \otimes \xi^B - \sigma^A \otimes \xi^B)) \\ &= \max_{0 \leq M^A, M^B \leq \mathbb{I}} \text{Tr}(M^A \otimes M^B(\rho^A - \sigma^A) \otimes \xi^B) \\ &= \max_{0 \leq M^A \leq \mathbb{I}} \text{Tr}(M^A(\rho^A - \sigma^A)) \max_{0 \leq M^B \leq \mathbb{I}} \text{Tr}(M^B \xi^B) \\ &= \max_{0 \leq M^A \leq \mathbb{I}} \text{Tr}(M^A(\rho^A - \sigma^A)) = D(\rho^A, \sigma^A). \end{aligned}$$

Using this fact and the triangle inequality we have

$$\begin{aligned} D(\rho^A \otimes \rho^B, \sigma^A \otimes \sigma^B) &\leq D(\rho^A \otimes \rho^B, \sigma^A \otimes \rho^B) + D(\sigma^A \otimes \rho^B, \sigma^A \otimes \sigma^B) \\ &= D(\rho^A, \sigma^A) + D(\rho^B, \sigma^B). \end{aligned}$$

□

2.6.2 Fidelity

Contrary to the quantum trace distance, quantum fidelity is a similarity measure. It also has a classical counterpart.

Definition 2.6.13 (Fidelity of Classical Distributions). *Let p and q be two distribution with labels in Γ . The fidelity of p and q is*

$$F(p, q) = \sum_{x \in \Gamma} \sqrt{p(x)} \sqrt{q(x)}.$$

The classical fidelity ranges from zero to one. It is zero if the distributions satisfy

$$p(x) > 0 \implies q(x) = 0$$

for all $x \in \Gamma$. It is one if and only if the distributions p and q are the same.

Definition 2.6.14 (Fidelity of Pure States). *Let $|\psi\rangle$ and $|\phi\rangle$ be two pure states. The fidelity of $|\psi\rangle$ and $|\phi\rangle$ is*

$$F(|\psi\rangle, |\phi\rangle) = |\langle\psi|\phi\rangle|.$$

Definition 2.6.15 (Fidelity of Mixed States). *Let ρ and σ be two mixed states. The fidelity of ρ and σ is*

$$F(\rho, \sigma) = \text{Tr}(\sqrt{\sqrt{\rho}\sigma\sqrt{\rho}}),$$

or equivalently

$$F(\rho, \sigma) = \left\| \sqrt{\rho}\sqrt{\sigma} \right\|_1.$$

The two expressions for the fidelity of mixed states in the previous definition are equivalent.

$$F(\rho, \sigma) = \left\| \sqrt{\rho}\sqrt{\sigma} \right\|_1 = \text{Tr}(\sqrt{A^\dagger A}),$$

where $A = \sqrt{\rho}\sqrt{\sigma}$. Consequently, we have $F(\rho, \sigma) = \text{Tr}(\sqrt{\sqrt{\sigma}\rho\sqrt{\sigma}})$.

Let $|\psi_\xi\rangle$ denote generically a purifications of ξ .

Theorem 2.6.16 (Uhlmann's Theorem). *Let ρ and σ be two density operators. The fidelity between them is*

$$F(\rho, \sigma) = \max_{|\psi_\sigma\rangle} |\langle\psi_\rho|\psi_\sigma\rangle|.$$

Proof. A purification of ρ and σ can be written as

$$\begin{aligned} |\psi_\rho\rangle &= \sum_i (\sqrt{\rho} \otimes I) |i\rangle |i\rangle, \text{ and} \\ |\psi_\sigma\rangle &= \sum_i (\sqrt{\sigma} V_1 \otimes V_2) |i\rangle |i\rangle, \end{aligned}$$

where U_i and V_i are unitaries and $\{|i\rangle\}$ is the eigenbasis of ρ .

$$\begin{aligned}
|\langle \psi_\rho | \psi_\sigma \rangle| &= \sum_i \sum_j \langle i | \langle i | (\sqrt{\rho} \otimes I) (\sqrt{\sigma} V_1 \otimes V_2) | j \rangle | j \rangle \\
&= \sum_i \sum_j \langle i | \sqrt{\rho} \sqrt{\sigma} V_1 | j \rangle \langle i | V_2 | j \rangle \\
&= \langle V_1^\dagger \sqrt{\sigma} \sqrt{\rho}, V_2 \rangle \\
&= \text{Tr}(\sqrt{\rho} \sqrt{\sigma} V_1 V_2)
\end{aligned}$$

For a square matrix A and a unitary U , we have $\text{Tr}(AU) \leq \text{Tr}((A^\dagger A)^{\frac{1}{2}})$ with equality only if U is the matrix in the polar decomposition of A . Therefore, we have

$$\max_{|\psi_\sigma\rangle} |\langle \psi_\rho | \psi_\sigma \rangle| = \text{Tr}(\sqrt{\sqrt{\rho} \sigma \sqrt{\rho}}).$$

□

Corollary 2.6.17. *Let ρ and σ be two density operators. It holds $0 \leq F(\rho, \sigma) \leq 1$.*

Proof. Uhlmann's Theorem stipulates that the fidelity is the overlap $|\langle \psi_\rho | \psi_\sigma \rangle|$ of two unitary vectors. □

Corollary 2.6.18. *Let ρ and σ be two density operators. Let $|\psi_\rho\rangle$ and $|\psi_\sigma\rangle$ be two fixed purifications of ρ and σ , respectively. The fidelity between ρ and σ is*

$$F(\rho, \sigma) = \max_{U^R} |\langle \psi_\rho | (I \otimes U^R) | \psi_\sigma \rangle|,$$

where U^R is a unitary acting in the reference system.

Proof. This result is immediate since purifications are related by unitaries applied on the reference system. Let $|\psi_\sigma\rangle$ be a fixed purification of σ . Any other purification of this mixed state can use a different basis for the reference systems. Therefore, for any $|\psi'_\sigma\rangle$, there is a unitary U^R such that $|\psi'_\sigma\rangle = (I \otimes U^R) |\psi_\sigma\rangle$. □

Lemma 2.6.19 (Fidelity Operational Interpretation (adapted from [74])). *Let ρ and σ be two density operators. The fidelity between them is*

$$F(\rho, \sigma) = \min_{\{M_x\}} F(p, q),$$

where the optimization is over all possible POVMs $\{M_x\}$ and p and q are the probability distributions associated to measuring ρ and σ with it, i.e. $p(x) = \text{Tr}(M_x \rho)$ and $q(x) = \text{Tr}(M_x \sigma)$.

Proof. Firstly, we show that the classical fidelity between probability distributions p and q arising from measuring ρ and σ , respectively, by any POVM can be used as an upper bound to $F(\rho, \sigma)$. Then, we show the existence of a POVM attaining equality.

The polar representation of operator $\sqrt{\sqrt{\sigma}\rho\sqrt{\sigma}}$ is $\sqrt{\rho}\sqrt{\sigma}U$ [74]. Fix a POVM $\{M_x\}$, we have

$$\begin{aligned}
 F(\rho, \sigma) &= \text{Tr}(\sqrt{\sqrt{\sigma}\rho\sqrt{\sigma}}) \\
 &= \text{Tr}(\sqrt{\rho}\sqrt{\sigma}U) \\
 &= \sum_x \text{Tr}(\sqrt{\rho}\sqrt{M_x}\sqrt{M_x}\sqrt{\sigma}U) \\
 &= \sum_x \langle \sqrt{\rho}\sqrt{M_x}, \sqrt{M_x}\sqrt{\sigma}U \rangle \\
 &\leq \sum_x \sqrt{\text{Tr}(M_x\rho)}\sqrt{\text{Tr}(M_x\sigma)} \text{ (Cauchy-Schwarz)} \\
 &= F(p, q),
 \end{aligned}$$

where $p(x) = \text{Tr}(M_x\rho)$ and $q(x) = \text{Tr}(M_x\sigma)$ for all x .

We proceed to show a POVM that attains equality. The Cauchy-Schwarz becomes an equality if and only if the vectors are colinear that is

$$\sqrt{\rho}\sqrt{M_x} = \alpha_x \sqrt{M_x}\sqrt{\sigma}U,$$

for a complex α_x .

From the polar decomposition $\sqrt{\rho}\sqrt{\sigma}U = \sqrt{\sqrt{\sigma}\rho\sqrt{\sigma}}$ and assuming ρ is invertible, we can write

$$\sqrt{\sigma}U = \sqrt{\rho}^{-1}\sqrt{\sqrt{\sigma}\rho\sqrt{\sigma}}.$$

Applying this result to the equality condition gives

$$\sqrt{M_x}(\mathbb{I} - \alpha_x P) = 0, \tag{2.3}$$

where $P = \sqrt{\rho}^{-1}\sqrt{\sqrt{\sigma}\rho\sqrt{\sigma}}\sqrt{\rho}^{-1}$. Let $P = \sum_x \lambda_x |x\rangle\langle x|$ be a spectral decomposition of M . Setting $M_x = |x\rangle\langle x|$ and $\alpha_x = \frac{1}{\lambda_x}$, we satisfy Eq. 2.3. Even if ρ is not invertible, the result follows from continuity [74]. \square

Contrary to the trace distance that is convex, fidelity is concave.

Claim 2.6.20 (Concavity). *Let ρ , σ , and ξ be density operators. For $\alpha \in [0, 1]$, it holds*

$$F(\alpha\rho + (1 - \alpha)\sigma, \xi) \geq \alpha F(\rho, \xi) + (1 - \alpha)F(\sigma, \xi).$$

The fidelity of three density operators ρ , σ , and ξ satisfy an inequality similar in flavor to the triangle inequality for the trace distance.

Lemma 2.6.21 (From [12]). *Let ρ , σ , and ξ be density operators, we have*

$$F(\rho, \xi)^2 + F(\xi, \sigma)^2 \leq 1 + F(\rho, \sigma).$$

2.6.3 Relating Measures

The fidelity and the trace distance are related to each other through the Fuchs-van de Graaf inequality. In face of to this inequality, there is no drawback in working exclusively with the trace distance or the fidelity, as a bound for one of them can be easily converted into a bound for the other.

Lemma 2.6.22 (Fuchs-van de Graaf inequality [39]).

$$1 - F(\rho, \sigma) \leq D(\rho, \sigma) \leq \sqrt{1 - F(\rho, \sigma)^2}.$$

Proof. Firstly, we show $D(\rho, \sigma) \leq \sqrt{1 - F(\rho, \sigma)^2}$. Let $|\psi_\rho\rangle$ and $|\psi_\sigma\rangle$ be purifications of ρ and σ , respectively, such that $F(\rho, \sigma) = |\langle\psi_\rho|\psi_\sigma\rangle|$. By Uhlmann's Theorem 2.6.16, we known that these purification exist. The trace distance for these pure states is

$$D(|\psi_\rho\rangle, |\psi_\sigma\rangle) = \sqrt{1 - |\langle\psi_\rho|\psi_\sigma\rangle|^2} = \sqrt{1 - F(\rho, \sigma)^2}.$$

Since the trace distance is contractible, we have the inequality $D(\rho, \sigma) \leq D(|\psi_\rho\rangle, |\psi_\sigma\rangle)$ which establishes the first claim.

Next, we proceed to show the inequality $1 - F(\rho, \sigma) \leq D(\rho, \sigma)$, as in [74]. From the operational interpretation, the fidelity of ρ and σ is

$$F(\rho, \sigma) = \sum_x \sqrt{p(x)q(x)},$$

where $p(x) = \text{Tr}(M_x \rho)$ and $q(x) = \text{Tr}(M_x \sigma)$ are the probabilities of measuring outcome x . Note that the following equality holds

$$\sum_x (\sqrt{p(x)} - \sqrt{q(x)})^2 = \sum_x p(x) + \sum_x q(x) - 2F(\rho, \sigma) \quad (2.4)$$

$$= 2(1 - F(\rho, \sigma)). \quad (2.5)$$

Using the inequality $|\sqrt{p(x)} - \sqrt{q(x)}| \leq |\sqrt{p(x)} + \sqrt{q(x)}|$, the left hand side of Eq. 2.5 can be bounded from above as

$$\sum_x (\sqrt{p(x)} - \sqrt{q(x)})^2 \leq \sum_x |\sqrt{p(x)} - \sqrt{q(x)}| |\sqrt{p(x)} + \sqrt{q(x)}| \quad (2.6)$$

$$= \sum_x |p(x) - q(x)| \quad (2.7)$$

$$= 2D(p, q) \leq 2D(\rho, \sigma). \quad (2.8)$$

Combining Eqs. 2.5 and 2.8, we conclude that

$$1 - F(\rho, \sigma) \leq D(\rho, \sigma).$$

□

Chapter 3

Ubiquity of Entanglement

Entanglement is an important and distinctive property of quantum mechanics. Roughly speaking, it captures the notion of quantum correlations which can not be described by any classical hidden variable theory. The quantum model provides great flexibility which makes the maximum allowed correlations stronger than any classical one. When we consider a composite quantum system with two subsystems A and B , even though the joint state might be a definite pure state, the reduced state on A and B might be mixed. This means that the joint state can not always be described by its parts alone. Note that this is in sharp contrast to the classical case in which the state of a composite register can always be described by its parts. The hardness of representing a quantum state classically is believed to be associated with how much entanglement it contains. One way to explore the computational power of quantum mechanics is to generate and use non-trivial entangled states as it is done in Shor's factoring algorithm [91]. Throughout the years, entanglement has found many application such as in cryptography with quantum keys distribution protocols and also in information theory providing new communication protocols. Entanglement also has physical implications to condensed matter physics, understanding the structure of the lowest energy state is central in the design of new materials.

A powerful approach to study entanglement is through the lens of computational complexity. The question $\text{QMA} = \text{QCMA}$ [5] can be viewed as asking whether highly entangled states are indeed more expressive for proof verification than states that are close to product admitting an efficient classical representation. In the context of multi-prover interactive proof system, if players share an entangled state prior to the execution of the protocol, intuitively it would be expected that they are able to collude and thus the systems would loose its soundness. Surprisingly, Vidick and Ito showed that this is not the case since $\text{NEXP} \subseteq \text{MIP}^*$ [57]. An upper bound for MIP^* is an open question. Another result in favor of the extra power provided by entanglement is a 5-prover entangled

quantum interactive proof with logarithm size query that can decide any QMA promise problem [38]. Nevertheless, there is a setting in which the lack of entanglement could be the source of an accrue expressive power, namely in $\text{QMA}(k)$ compared to QMA [65].

Quantum mechanics, and more specifically entanglement, was also criticised by Einstein, Podolsky, and Rosen in what became known as the EPR paradox [33]. Suppose two particles A and B were prepared in the valid quantum state $|\phi^+\rangle = \frac{1}{\sqrt{2}}(|00\rangle^{AB} + |11\rangle^{AB})$ and separated to an arbitrarily long distance. This state $|\phi^+\rangle$ became known as the EPR pair. If we measure one of the particles, the other one simultaneously collapses to the same state. For appropriate large distances this phenomenon would violate the limits imposed by the speed of light, thus constituting a paradox with respect to relativity. In Einstein's words this was a "spooky action at a distance". For him, somehow, when the state was prepared some hidden variable that was not present in the model was set and it could be used to explain this phenomenon. This hypothesis of hidden variables was experimentally ruled out by violations of the Bell inequalities [86].

This chapter is a compilation of basic results regarding entanglement. Despite their simplicity, they already show how this property is a distinctive feature of quantum mechanics. Firstly, entanglement is formally defined and the Bell states are introduced. The maximally entangled states are presented with some interesting associated properties. Pure bipartite states admit a neat representation denoted Schmidt decomposition. Building on this decomposition, it is shown that all mixed states can be viewed as a pure states that are entangled with the environment. Next, the CHSH [25] and Magic Square games are presented. They are important witnesses of the extra power of entangled strategies. Moreover, two simple information processing tasks are explained: quantum teleportation and superdense coding. Finally, the Choi-Jamiołkowski representation of a quantum channel can be understood in terms of maximally entangled states.

3.1 Entangled State Definition

Entanglement of a composite system is defined with respect to a partition into smaller systems. Given a composite quantum system and a bipartition into subsystems A and B , there are ways of measuring the entanglement across this cut such as the Schmidt Rank or the entropy of entanglement. We provide separate definitions of entangled state depending whether its pure or mixed. A pure state is said to be entangled if it meets the following definition.

Definition 3.1.1 (Entangled Pure State). *The pure state $|\xi\rangle^{AB}$ is **entangled** if and only if it can **not** be written as $|\xi\rangle^{AB} = |i\rangle^A \otimes |j\rangle^B$ where $|i\rangle^A$ and $|j\rangle^B$ are pure states on subsystems A and B , respectively.*

The problem of determining if a pure state is entangled or not, given its classical description, is in P. As we show later, the Schmidt decomposition can be used for that end. For a mixed state ρ^{AB} , if it is not entangled across the cut A and B , we say it is separable.

Definition 3.1.2 (Separable State). *The state ρ^{AB} is separable if there is probability vector p such that $\rho^{AB} = \sum_i p(i) \rho_i^A \otimes \rho_i^B$.*

With the separable state definition, we are ready to state the entangled mixed state definition.

Definition 3.1.3 (Entangled Mixed State). *The mixed state ρ^{AB} is **entangled** if and only if it is not separable.*

Deciding if a state is close to separable within an inverse polynomial error in the dimension is an NP-hard problem [42]. This hardness result is the reason why a simple and easily verifiable characterization of mixed entangled states are unlikely to exist. For this reason, many practical heuristics to test entanglement were created. The Positive Partial Transpose (PPT) is an example of this kind of test. It consists in transposing only one of the subsystems and checking if the resulting state is still positive. It is clear that all separable states pass this test. However, due to the hardness associated to the task, it is not fruitful to expect that all entangled states fail any polynomial time test like this one.

The convex combination of two separable states $\rho^{AB} = \sum_{i=0}^{N-1} p(i) \rho_i^A \otimes \rho_i^B$ and $\sigma^{AB} = \sum_{j=0}^{M-1} q(j) \sigma_j^A \otimes \sigma_j^B$ results in a state ξ^{AB} of the form

$$\xi^{AB} = \alpha \rho^{AB} + (1 - \alpha) \sigma^{AB} = \sum_{k=0}^{N+M-1} r(k) \xi_k^A \otimes \xi_k^B,$$

where $r(k) = \alpha p(i)$ and $\xi_k^A \otimes \xi_k^B = \rho_k^A \otimes \rho_k^B$ for $k < N$, and $r(k) = (1 - \alpha)q(k)$ and $\xi_k^A \otimes \xi_k^B = \sigma_k^A \otimes \sigma_k^B$, otherwise. The state ξ^{AB} is clearly a separable state and thus the set of separable states is a convex set. Optimizing over convex sets are in many cases computationally efficient (*i.e.* it is in P) as the sets encountered in linear programming. However, the problem of optimizing a positive operator M satisfying $0 \leq M \leq \mathbb{I}$ over the separable states is NP-hard up to an inverse polynomial error in the dimension.

Definition 3.1.4 (BSS(ϵ) [20]). *In the Best Separable State problem the input is an Hermitian operator M on systems A and B , and the goal is to maximize the expression*

$$\max_{\rho^{AB} \in \text{SEP}(A,B)} \langle M, \rho^{AB} \rangle,$$

within an additive error ϵ where $\text{SEP}(A,B)$ is the set of separable state across the systems A and B .

In the definition of separable states, the only allowed correlations are given by classical probability distributions. It is important to point out that classical correlation can be arbitrarily extended. For instance, the state $\rho^{AB} = \sum_i p(i) \rho_i^A \otimes \rho_i^B$ can be extended to $\rho^{AB\dots B} = \sum_i p(i) \rho_i^A \otimes \rho_i^B \otimes \dots \otimes \rho_i^B$. On the other hand, quantum correlations can not be arbitrarily extended as quantum de Finetti theorems show [30]. Entanglement is in a certain sense “monogamous”, that is, a quantum system can not be entangled with many others at the same time.

3.2 Bell States

Among the simplest entangled states are the Bell states. They are just a coherent superposition of two computational basis states as shown bellow

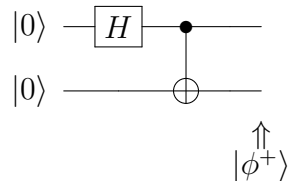
$$\begin{aligned} |\phi^+\rangle &= \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle), \\ |\phi^-\rangle &= \frac{1}{\sqrt{2}}(|00\rangle - |11\rangle), \\ |\psi^+\rangle &= \frac{1}{\sqrt{2}}(|01\rangle + |10\rangle), \text{ and} \\ |\psi^-\rangle &= \frac{1}{\sqrt{2}}(|01\rangle - |10\rangle). \end{aligned}$$

These state are ubiquitous in quantum computing. Even tough $|\phi^+\rangle$ and $|\phi^-\rangle$ are entangled states, a uniform coherent superposition of them is just the unentangled state

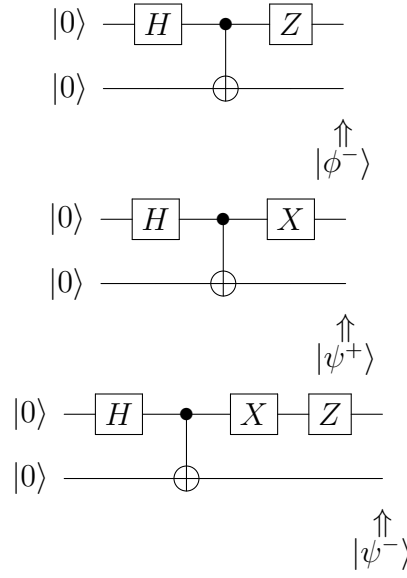
$$\frac{1}{\sqrt{2}}|\phi^+\rangle + \frac{1}{\sqrt{2}}|\phi^-\rangle = |00\rangle = |0\rangle \otimes |0\rangle.$$

This is not at all a surprising result since the bell states forms an orthonormal basis for \mathbb{C}^4 . Therefore, a combination of them must be able to generate any state in this space, including those that are not entangled.

The state $|\phi^+\rangle$, also known as an EPR pair, can be created from $|0\rangle^A|0\rangle^B$ by applying the Hadamard gate to A and a controlled not gate (CNOT) using A as control an B as a target. This sequence of operations is depicted in the circuit below.



It is the CNOT gate that does the job of entangling the systems A and B . The remaining Bell states can be generated by similar circuits with the addition of the Pauli operators X and Z . The circuits for generating them are shown next.



It is important to observe that to create entanglement the two systems must be brought together. Using only local operations and classical communication (LOCC), it is not possible to create more than classical correlations.

3.3 Maximally Entangled States

The maximally entangled states constitute a family of states generalizing EPR pairs for arbitrary dimensions. It has interesting properties, some of which we cover next. For this reason, they have many applications. For a system of two d -dimensional qudits, the maximally entangled state is

$$|\psi\rangle^{AB} = \sum_{i=0}^{d-1} \frac{1}{\sqrt{d}} |i\rangle^A |i\rangle^B.$$

For a $2n$ -qubit system, the maximally entangled state is

$$|\psi\rangle^{AB} = \sum_{i=0}^{2^n-1} \frac{1}{\sqrt{2^n}} |i\rangle^A |i\rangle^B.$$

This state can be generated by creating n EPR pairs and grouping their first and second halves into systems A and B , respectively. As a consequence, such states can be generated in linear time in the parameter n . For simplicity, we make as our reference the maximally entangled state to be a $2n$ -qubit system.

The reduced state on system A (or B) of a maximally entangled state is the totally mixed state $\frac{1}{2^n} I$. One interesting property of $|\psi\rangle^{AB}$ occurs when system A (or B) is measured according to a real orthonormal basis $\{|j\rangle\}$ satisfying $\langle j|i\rangle \in \mathbb{R}$ for all $|i\rangle \in \{|i\rangle\}$

and $|j\rangle \in \{|j\rangle\}$. If the outcome is $|j\rangle^A$, then system B also collapses to state $|j\rangle^B$. This can be seen by rewriting the maximally entangled state in the new basis as

$$\begin{aligned}
|\psi\rangle^{AB} &= \frac{1}{\sqrt{2^n}} \sum_{i=0}^{2^n-1} |i\rangle^A |i\rangle^B \\
&= \frac{1}{\sqrt{2^n}} \sum_{i=0}^{2^n-1} \left(\sum_{j=0}^{2^n-1} \langle j|i\rangle |j\rangle^A \right) \left(\sum_{j'=0}^{2^n-1} \langle j'|i\rangle |j'\rangle^B \right) \\
&= \frac{1}{\sqrt{2^n}} \sum_{i=0}^{2^n-1} \sum_{j=0}^{2^n-1} \sum_{j'=0}^{2^n-1} \langle j|i\rangle \langle j'|i\rangle |j\rangle^A |j'\rangle^B \\
&= \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} \sum_{j'=0}^{2^n-1} \left(\sum_{i=0}^{2^n-1} \langle j|i\rangle \langle i|j'\rangle \right) |j\rangle^A |j'\rangle^B \\
&= \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} |j\rangle^A |j\rangle^B.
\end{aligned}$$

The state $|\psi\rangle^{AB}$ exhibits another useful property. When a unitary operation U is applied to just one of the subsystems, say A , the end state is the same as if U^t were applied to B . This is shown in the computation

$$\begin{aligned}
(U^A \otimes I^B)|\psi\rangle &= \frac{1}{\sqrt{2^n}} \sum_{i=0}^{2^n-1} U^A |i\rangle^A |i\rangle^B \\
&= \frac{1}{\sqrt{2^n}} \sum_{i=0}^{2^n-1} \left(\sum_{j=0}^{2^n-1} U_{ji} |j\rangle^A \right) |i\rangle^B \\
&= \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} |j\rangle^A \sum_{i=0}^{2^n-1} U_{ji} |i\rangle^B \\
&= \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} |j\rangle^A \left(\sum_{i=0}^{2^n-1} U_{ji} |i\rangle^B \right) \\
&= \frac{1}{\sqrt{2^n}} \sum_{j=0}^{2^n-1} |j\rangle^A U^t |i\rangle^B \\
&= (I^A \otimes \{U^t\}^B) \frac{1}{\sqrt{2^n}} |\psi\rangle.
\end{aligned}$$

Note that if U is unitary U^t is unitary as well, what is confirmed by the calculation

$$\begin{aligned}
(U^t)^\dagger U^t &= U^\star U^t = \sum_i \sum_j \left(\sum_k U_{ik}^\star U_{kj}^t \right) e_{ij} \\
&= \sum_i \sum_j \left(\sum_k U_{ik}^\star U_{jk} \right) e_{ij} \\
&= \sum_i \sum_j \delta_{ij} e_{ij} = I,
\end{aligned}$$

where e_{ij} are the canonical basis. This means that when Alice has system A and Bob system B , Alice can “remotely” apply any unitary to Bob’s system, and vice-versa.

3.4 Schmidt Decomposition

There is a very convenient way of writing a pure bipartite state $|\psi\rangle^{AB}$ on systems A and B , known as a Schmidt Decomposition (SD). This representation helps understanding the structure of the state across the cut of A and B .

Lemma 3.4.1 (Schmidt Decomposition). *Any pure state $|\psi\rangle^{AB}$ can be written as $|\psi\rangle^{AB} = \sum_{k=0}^{\min\{\dim(A), \dim(B)\}-1} \lambda_k |k\rangle^A |k\rangle^B$ where $\lambda_k \geq 0$ and $\{|k\rangle^A\}, \{|k\rangle^B\}$ are bases for A and B , respectively.*

Proof. The state $|\psi\rangle^{AB}$ can be written as $|\psi\rangle^{AB} = \sum_{i=0}^{\dim(A)-1} \sum_{j=0}^{\dim(B)-1} \alpha_{ij} |i\rangle^A |j\rangle^B$ where $\sum_{ij} |\alpha_{ij}|^2 = 1$ and $\{|i\rangle^A\}, \{|j\rangle^B\}$ are arbitrary orthonormal bases for systems A and B , respectively. From this representation, we construct an operator M whose M_{ij} entry is equal to α_{ij} . Applying the singular value decomposition (SVD) to M , we have

$$M = U \Sigma V,$$

where U and V are unitaries, $\Sigma_{ij} = 0$ for $i \neq j$, and $\Sigma_{ii} = \lambda_i \geq 0$. The matrix Σ is a $\dim(A) \times \dim(B)$ matrix, and so there are at most $\min\{\dim(A), \dim(B)\}$ non zero λ_k values. Using this new representation of M , a coefficient α_{ij} can be computed as

$$\alpha_{ij} = \sum_k U_{ik} \lambda_k V_{kj}.$$

The whole state $|\psi\rangle^{AB}$ can be rewritten in terms of this identity for α_{ij} as

$$\begin{aligned} \sum_i \sum_j \sum_k U_{ik} \lambda_k V_{kj} |i\rangle |j\rangle &= \sum_k \lambda_k \left(\sum_i U_{ik} |i\rangle \right) \left(\sum_j V_{kj} |j\rangle \right) \\ &= \sum_k \lambda_k |k\rangle^A |k\rangle^B \end{aligned}$$

where $\{|k\rangle^A = \sum_i U_{ik} |i\rangle\}$ and $\{|k\rangle^B = \sum_j V_{kj} |j\rangle\}$. The set $\{|k\rangle^A\}$ forms an orthonormal basis as shown next

$$\langle k' | k \rangle^A = \sum_{i'} \sum_i U_{i'k'}^\dagger U_{ik} \langle i' | i \rangle = \sum_i U_{ik'}^\dagger U_{ik} = \delta_{k'k}.$$

This expression holds because the columns of a unitary matrix forms an orthonormal basis.

The set $\{|k\rangle^B\}$ is also orthonormal. Each vector $|k\rangle^B$ is just the application of V to $|j\rangle$. Therefore, the inner product of $|k'\rangle$ and $|k\rangle$ is

$$\langle k'|k\rangle^B = \langle j'|V^\dagger V|j\rangle = \delta_{j'j}.$$

□

As shown in the previous proof, the SD is a consequence of the singular value decomposition. This tool provides a canonical way to represent bipartite pure states. Unfortunately, there is no general decomposition for more than two subsystem. This is one of the reasons why bipartite entanglement is reasonably well understood whereas the multipartite entanglement is not. The number of non-zero coefficients in the SD is known as the Schmidt Rank (SR), and it is used as a simple measure of entanglement across a fixed cut. This number can also be used to characterize entanglement in pure states as follows.

Definition 3.4.2. *A state $|\psi\rangle$ is entangled if and only if its Schmidt Rank is greater than one.*

A simple, yet useful, application of the SD is to calculate the reduced density matrix of a pure state. Let $\sum_k \lambda_k |k\rangle^A |k\rangle^B$ be the SD of $|\psi\rangle^{AB}$. Then the reduced state obtained by tracing out B is just $\text{Tr}_B(|\psi\rangle\langle\psi|^{AB}) = \sum_k \lambda_k^2 |k\rangle\langle k|^A$.

3.5 Purification and Environment

Entanglement also provides a mathematical abstraction used to view a mixed state ρ on system A as a pure state $|\psi\rangle^{AE}$ in a larger system composed of A and a reference system E . Without loss of generality, the dimension of E is at most the dimension of A . In a certain sense, to prove this result we apply the Schmidt decomposition backwards. Let $\rho^A = \sum_i \lambda_i |i\rangle\langle i|$ be the spectral decomposition of ρ^A . Now consider the state $|\psi\rangle^{AE} = \sum_i \sqrt{\lambda_i} |i\rangle^A |i\rangle^E$. This representation of $|\psi\rangle^{AE}$ is clearly in the Schmidt decomposition form, and this makes the equality $\text{Tr}_E(|\psi\rangle\langle\psi|^{AE}) = \rho^A$ evident. The state $|\psi\rangle^{AE}$ is called a purification of ρ^A . Note that a purification is not necessarily unique since any basis for the reference systems could be used. This concept of purification favors the interpretation that a non-pure, *i.e.* mixed state, is entangled with its environment.

3.6 XOR Games

The importance of games can be attributed to their appearance in a variety of contexts. In complexity theory, many classes are defined in terms of games and the famous PCP

Theorem has an equivalent game formulation. In quantum computing, games are useful to shed light on the power and limitations of entanglement. A subclass of games denoted **XOR** games is simple, and yet it captures several interesting aspects of the superclassical correlation propelled by entanglement. The exposition here is based on the lecture notes of Thomas Vidick along with the references therein [95] which provide an excellent exposition to games.

In full generality, a classical k -player game can be defined as follows.

Definition 3.6.1 (k -player Game). *A k -player Game $G = (Q, A, \pi, V)$ consists of a set of questions $Q = \{Q_1, \dots, Q_n\}$, a set of answers $A = \{A_1, \dots, A_m\}$, a distribution $\pi : Q^k \rightarrow [0, 1]$, and a weight function $V(a_1, \dots, a_k | q_1, \dots, q_k) : A^k \times Q^k \rightarrow [0, 1]$ such that the value of the game $\omega(G)$ is*

$$\omega(G) = \sup_{S \in (S_1, \dots, S_k)} \sum_{(q_1, \dots, q_k) \in Q^k} \pi(q_1, \dots, q_k) V(S_1(q_1), \dots, S_k(q_k) | q_1, \dots, q_k),$$

where $S_i : Q \rightarrow A$ is the strategy of the i^{th} player.

The previous game definition can be extended to allow shared randomness among the players. In this case, the value ω is “the randomized value of the game”, $\omega_r(G)$, and is defined as

$$\omega_r(G) = \sup_{S \in (S_1, \dots, S_k), R, \mu} \sum_{(q_1, \dots, q_k) \in Q^k} \pi(q_1, \dots, q_k) \sum_{r \in \{0,1\}^R} \mu(r) V(S_1(q_1, r), \dots, S_k(q_k, r) | q_1, \dots, q_k),$$

where $S_i : Q \times \{0, 1\}^R \rightarrow A$ is the strategy of the i^{th} player, μ is a probability distribution, and r the random string. By convexity, it holds that

$$\sum_{r \in \{0,1\}^R} \mu(r) V(S_1(q_1, r), \dots, S_k(q_k, r) | q_1, \dots, q_k) \leq \max_{r \in \{0,1\}^R} V(S_1(q_1, r), \dots, S_k(q_k, r) | q_1, \dots, q_k).$$

Denote by r' a value of r that maximizes the *rhs* of the previous inequality. Replacing the function S_i by $S'_i(q) = S_i(q, r') : Q \rightarrow A$, we have a deterministic strategy whose value is at least as good as the randomized one, implying that $\omega(G) \geq \omega_r(G)$. Following a similar reasoning, private randomness also does not give any advantage over deterministic strategies.

In contrast to randomness that is useless for the players, keeping the referee and the messages classical, but allowing the players to share an arbitrary entangled state $|\psi\rangle$ may lead to non-trivial variations of games. In this case, the i^{th} player strategy $S_i = \{M^{Q_1}, \dots, M^{Q_n}\}$ is a set of POVMs where each $M^q = \{M_{A_1}^q, \dots, M_{A_1}^q\}$ is indexed

by the received query q from the referee. The value of the game is the entangled value, $\omega^*(G)$. It is formally defined as

$$\omega^*(G) = \sup_{S \in (S_1, \dots, S_k), |\psi\rangle} \sum_{(q_1, \dots, q_k) \in Q^k} \pi(q_1, \dots, q_k) \sum_{(a_1, \dots, a_k) \in A^k} V(a_1, \dots, a_k | q_1, \dots, q_k) \langle \psi | M_{a_1}^{q_1} \otimes \dots \otimes M_{a_k}^{q_k} | \psi \rangle.$$

One of the potential difficulties in evaluating $\omega^*(G)$ is that the quantum state $|\psi\rangle$ may live in an arbitrarily large Hilbert space.

Definition 3.6.2 (Two-player **XOR** Game). *A two-player **XOR** Game G is a restricted type of game where $A = \{0, 1\}$ and the weight function is $V(a_1, a_2 | q_1, q_2) = f_{q_1, q_2}(a_1 \oplus a_2)$, such that the value of the game $\omega(G)$ is*

$$\omega(G) = \sup_{S \in (S_1, S_2), |\psi\rangle} \sum_{(q_1, q_2) \in Q^2} \pi(q_1, q_2) \sum_{(a_1, a_2) \in A^2} f_{q_1, q_2}(a_1 \oplus a_2) \langle \psi | M_{a_1}^{q_1} \otimes M_{a_2}^{q_2} | \psi \rangle, \quad (3.1)$$

where $S_i = \{M^{Q_1}, \dots, M^{Q_n}\}$ is the strategy of the i^{th} player.

Note that the strategy of an entangled **XOR** game consists only of binary POVMs.

3.6.1 CHSH Game

The superclassical strength of quantum entanglement correlation can be studied in the context of games. The CHSH is one of the most well known quantum games, it was named after its authors Clauser, Horne, Shimony, and Holt [25]. It is simple, and yet capable to prove the superiority of quantum correlations. In this game, two players, Alice and Bob, are given by a referee one classical bit each, denoted x and y , and chosen uniformly at random. Alice must reply with a classical bit a and Bob with a classical bit b . Then, the referee accepts if and only if $a \oplus b = x \wedge y$. The goal of the players is to maximize the referee's acceptance probability. Note that the players will try to approximate the value of the logical **AND** of x and y with the logical **XOR** of their answers. This game is formally defined next.

- 1 Referee chooses uniformly at random $(x, y) \in \{(0, 0), (0, 1), (1, 0), (1, 1)\}$.
- 2 Referee sends x to Alice and y to Bob which reply with answers a and b , respectively.
- 3 The players win if and only if $x \wedge y = a \oplus b$.

Algorithm 1: CHSH Game.

x	y	$x \wedge y$
0	0	0
0	1	0
1	0	0
1	1	1

A simple inspection of the logical **AND** truth table of x and y , shows that if Alice and Bob always reply with zero the referee accepts in $\frac{3}{4}$ of the cases.

A natural question is whether it is possible to achieve a higher probability using a classical deterministic strategy. To analyze it, let a_x and b_y be Alice and Bob answers upon receiving x and y , respectively. Suppose it holds that $a_x \oplus b_y$ is equal to $x \wedge y$ for all four cases, that is,

$$\begin{aligned} a_0 \oplus b_0 &= 0 \wedge 0, \\ a_0 \oplus b_1 &= 0 \wedge 1, \\ a_1 \oplus b_0 &= 1 \wedge 0, \text{ and} \\ a_1 \oplus b_1 &= 1 \wedge 1. \end{aligned}$$

Summing these four equations in \mathbb{F}^2 , we have a contradiction $0 = 1$. Therefore, it is not possible to accept more than $\frac{3}{4}$ of the inputs, proving that our initial strategy was optimum. Due to convexity, a probabilistic classical strategy is not capable of improving the acceptance probability.

Now, we consider the case in which the players share a predefined entangled state before the beginning of the game. We show that a specific quantum strategy outperforms any possible classical strategy, that is, $\omega^*(CHSH) > \omega(CHSH)$. The entangled state Alice and Bob share is just an EPR pair $|\phi^+\rangle = \frac{1}{\sqrt{2}}(|0\rangle^A|0\rangle^B + |1\rangle^A|1\rangle^B)$. To wisely use the quantum correlation, the players measure their halves using a rotated basis that contains states of the form

$$\begin{aligned} |\psi_0^z(\theta_z)\rangle &= \cos(\theta_z)|0\rangle + \sin(\theta_z)|1\rangle \text{ and} \\ |\psi_1^z(\theta_z)\rangle &= -\sin(\theta_z)|0\rangle + \cos(\theta_z)|1\rangle, \end{aligned}$$

which corresponds to rotations on the plane spanned by $\{|0\rangle, |1\rangle\}$. Depending on the classical bit x received, Alice measures in the basis $\{|\psi_a^x(\theta_x)\rangle\}_{a \in \{0,1\}}$ where $\theta_0 = 0$ and $\theta_1 = \frac{\pi}{4}$, returning the outcome given by a . Similarly, depending on y Bob measures in the basis $\{|\psi_b^y(\theta_y)\rangle\}_{b \in \{0,1\}}$ where $\theta_0 = \frac{\pi}{8}$ and $\theta_1 = -\frac{\pi}{8}$, returning b . It is much easier to understand why this strategy makes use of quantum correlation from a graphical plot of these basis as it is shown next.

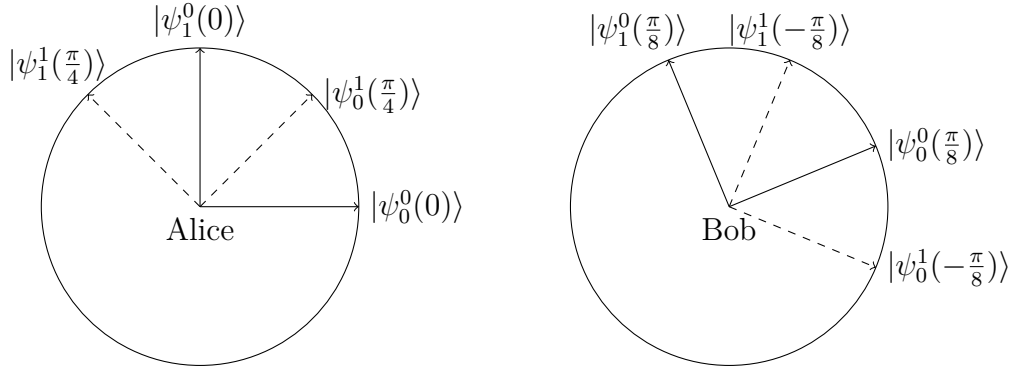


Figure 3.1: Alice's strategy is on the left. Upon receiving bit x , she measures in the $\{|\psi_a^x(\theta_x)\rangle\}_{a \in \{0,1\}}$ basis and returns a . Analogously, Bob's strategy is on the right. Upon receiving bit y , he measures in the $\{|\psi_b^y(\theta_y)\rangle\}_{b \in \{0,1\}}$ basis and returns b .

Since each player measures its own half of the EPR pair, their measurements commute. To simplify the analysis, we can assume that Alice always measures first. Recall that the EPR is a maximally entangled state. Thus after Alice's measurement Bob's state collapses to the same state she measured. For each input, we show in the table below the outcomes for Alice and Bob that would cause the referee to accept.

x	y	$x \wedge y$	Alice	Bob
0	0	0	$ \psi_0^0(\theta_0)\rangle = \cos(0) 0\rangle + \sin(0) 1\rangle$	$ \psi_0^0(\theta_0)\rangle = \cos(\frac{\pi}{8}) 0\rangle + \sin(\frac{\pi}{8}) 1\rangle$
			$ \psi_1^0(\theta_0)\rangle = -\sin(0) 0\rangle + \cos(0) 1\rangle$	$ \psi_1^0(\theta_0)\rangle = -\sin(\frac{\pi}{8}) 0\rangle + \cos(\frac{\pi}{8}) 1\rangle$
0	1	0	$ \psi_0^0(\theta_0)\rangle = \cos(0) 0\rangle + \sin(0) 1\rangle$	$ \psi_0^1(\theta_1)\rangle = \cos(-\frac{\pi}{8}) 0\rangle + \sin(-\frac{\pi}{8}) 1\rangle$
			$ \psi_1^0(\theta_0)\rangle = \sin(0) 0\rangle + \cos(0) 1\rangle$	$ \psi_1^1(\theta_1)\rangle = -\sin(-\frac{\pi}{8}) 0\rangle + \cos(-\frac{\pi}{8}) 1\rangle$
1	0	0	$ \psi_0^1(\theta_1)\rangle = \cos(\frac{\pi}{4}) 0\rangle + \sin(\frac{\pi}{4}) 1\rangle$	$ \psi_0^0(\theta_0)\rangle = \cos(\frac{\pi}{8}) 0\rangle + \sin(\frac{\pi}{8}) 1\rangle$
			$ \psi_1^1(\theta_1)\rangle = -\sin(\frac{\pi}{4}) 0\rangle + \cos(\frac{\pi}{4}) 1\rangle$	$ \psi_1^0(\theta_0)\rangle = -\sin(\frac{\pi}{8}) 0\rangle + \cos(\frac{\pi}{8}) 1\rangle$
1	1	1	$ \psi_0^1(\theta_1)\rangle = \cos(\frac{\pi}{4}) 0\rangle + \sin(\frac{\pi}{4}) 1\rangle$	$ \psi_1^1(\theta_1)\rangle = -\sin(-\frac{\pi}{8}) 0\rangle + \cos(-\frac{\pi}{8}) 1\rangle$
			$ \psi_1^1(\theta_1)\rangle = -\sin(\frac{\pi}{4}) 0\rangle + \cos(\frac{\pi}{4}) 1\rangle$	$ \psi_0^1(\theta_1)\rangle = \cos(-\frac{\pi}{8}) 0\rangle + \sin(-\frac{\pi}{8}) 1\rangle$

Table 3.1: CHSH successful Pair of states.

It is a simple exercise to check that the square root norm of the inner product of Alice and Bob states, in all the above cases, is $\cos^2(\frac{\pi}{8}) \approx 0.854 > \frac{3}{4} = \omega(CHSH)$. Again, a graphical view of this pair of vector make it much easier to see the angles of $\frac{\pi}{8}$.

3.6.2 MAXCUT game

As the CHSH is perhaps the most famous quantum game, MAXCUT is its classical counterpart in terms of notoriety. The MAXCUT problem asks for the number of edges in a maximum cut. It is known to be NP-hard, and so its game will also be NP-hard

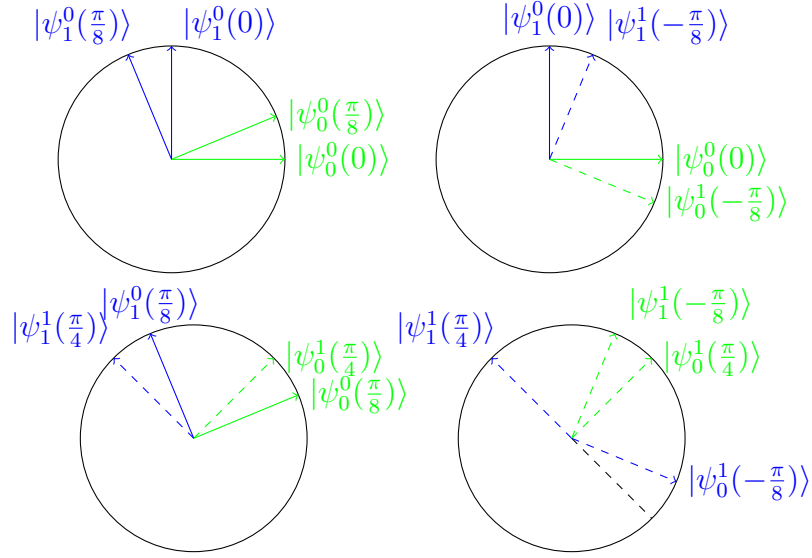


Figure 3.2: In each circle, pairs of vectors from the same color correspond to answers that cause the referee to accept. In all cases, the absolute angle is $\frac{\pi}{8}$ implying that the referee accepts with probability $\cos^2(\frac{\pi}{8})$.

to approximate within a certain factor. Before we introduce the MAXCUT game, we adopt the following convenient notation. For a binary value b , we map zero to one and one to minus one as in $(-1)^b$. With this notation the logical operation **XOR** becomes simply a multiplication. This game is also a two player game whose input is a graph $G = (V, E)$. Alice and Bob strategies consist in a boolean assignment to vertices V which we denote by $(x_i)_{i \in V}$ and $(y_j)_{j \in V}$, respectively. The referee selects one edge $e = (i, j)$ from E uniformly at random, and then with equal probability perform one of the following actions:

- (i) asks Alice and Bob the value of i ;
- (ii) asks Alice and Bob the value of j ;
- (iii) asks the value of i for Alice and j for Bob; and
- (iv) asks the value of j for Alice and i for Bob.

For cases (i) and (ii), the referee accepts if and only if their answers are equal. For the other two cases, the referee accepts if and only if their answers are different. We can think about the player's strategies as determining a bipartition (S, \bar{S}) of G 's vertices V , where 1 indicates that the vertex is in S , and -1 indicates otherwise. If the two strategies were the same, the two last cases would estimate the number of edges in the cut. However, the

players may collude, making the two first tests indispensable to enforce consistency of their strategy. Apart from our notation, this game is indeed a **XOR** game and its value is

$$\begin{aligned}
\omega(\text{MAXCUT}) &= \max_{(x_i)_i} \max_{(y_j)_j} \mathbf{E}_{e=(i,j)} \left[\frac{1}{4} \left(\frac{1+x_i y_i}{2} + \frac{1-x_i y_j}{2} + \frac{1-x_j y_i}{2} + \frac{1+x_j y_j}{2} \right) \right] \\
&= \frac{1}{2} + \max_{(x_i)_i} \max_{(y_j)_j} \mathbf{E}_{e=(i,j)} \left[\frac{(x_i - x_j)(y_i - y_j)}{8} \right] \\
&\leq \frac{1}{2} + \frac{1}{2} \max_{(x_i)_i} \mathbf{E}_{e=(i,j)} \left[\left(\frac{x_i - x_j}{2} \right)^2 \right] \\
&= \frac{1}{2} + \frac{1}{2} \frac{\#\text{MAXCUT}}{\#\text{EDGES}}.
\end{aligned}$$

A consequence of the previous expression is the following theorem.

Theorem 3.6.3. *The value ω for an **XOR** game is NP-hard problem to approximating within a $(1 + \frac{1}{\text{poly}(n)})$ factor of the optimum where n is the input size.*

This result follows from MAXCUT being a NP-hard and approximation the value of the game within this error can be used to recover the number of edges in the maximum cut.

3.6.3 Magic Square

The Magic Square (MS) from Mermin-Peres is one of the earliest examples of a game showing that by sharing entanglement players can always fool the referee. This game raised the question that entanglement might be always used for the player's advantage, hurting the soundness of multi-player games. The result $\text{NEXP} \subseteq \text{MIP}^*$, where MIP^* is the classical MIP in which players start with an arbitrary entangled state, showed that this is not necessarily the case. Moreover, it is not known any upper bound on MIP^* . It might be the case that entanglement is indeed a powerful resource in the referee's favor instead of a proponent to players collusion.

The MS is also a two-player **XOR** game. The two players must convince the referee that a 3-by-3 table of boolean variables (x_{ij}) can be filled in a way that the parity of each row and column is determined as indicated below. Note that we have again used the notation $(-1)^b$ for boolean values in $\{0, 1\}$.

The game can be defined as follows 2.

Similarly to the CHSH game, it is possible to use a parity argument to rule out the possibility of fixed assignment to the boolean variables in the MS table that makes the referee accept with probability 1. To see this, for the sake of contradiction suppose such an assignment (x_{ij}) exists. If we multiply the constraints, that is, take the logical **XOR**

x_{11}	x_{12}	x_{13}	1
x_{21}	x_{22}	x_{23}	1
x_{31}	x_{32}	x_{33}	1
1	1	-1	

Table 3.2: Mermim-Peres magic square parity rule for rows and columns.

Referee selects with probability $\frac{1}{2}$ row or column.

Referee chooses

- $i \in \{1, 2, 3\}$ which indexes the row or column,
- $j \in \{1, 2, 3\}$ which indexes a variable $y = x_{ij}$ or $y = x_{ji}$ in the chosen row or column.

Referee asks

- Alice the value of all variables in the selected row or line,
- Bob the value of y .

Referee accepts if and only if

- the parity of Alice answer is correct, and
- Alice and Bob answers on y are equal.

Algorithm 2: Magic Square Game.

$$\begin{aligned} x_{11}x_{12}x_{13} &= 1, x_{21}x_{22}x_{23} = 1, x_{31}x_{32}x_{33} = 1, \\ x_{11}x_{21}x_{31} &= 1, x_{12}x_{22}x_{32} = 1, x_{13}x_{23}x_{33} = -1, \end{aligned}$$

we get $1 = -1$ because each variable will appear squared allowing them to be replaced by 1. Without loss of generality, we can assume that Alice's answers always have the correct parity. Among the six possibilities of rows and columns, she must lie in at least the value of one of its variables. The referee will detect this inconsistency with probability $\frac{1}{18}$. Thus $\omega(MS) \leq \frac{17}{18}$. Surprisingly, allowing the players to share an entangled state, the following lemma holds.

Lemma 3.6.4. $\omega^*(MS) = 1$.

Before presenting an optimum strategy, recall the Pauli matrices and their spectral decomposition

$$\begin{aligned} X &= \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}, \\ Y &= \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix} = \frac{1}{2} \begin{pmatrix} 1 & 1 \\ i & -i \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 1 & -i \\ 1 & i \end{pmatrix} \text{ and} \\ Z &= \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}. \end{aligned}$$

The eigenvalues of X , Y , and Z are in $\{1, -1\}$. Thus they square to the identity: $X^2 = Y^2 = Z^2 = \mathbb{I}$. Furthermore, their product satisfy

$$\begin{aligned} XY &= iZ, YX = -iZ, \\ XZ &= -iY, ZX = iY, \\ YZ &= iX, ZY = -iX. \end{aligned}$$

From these identities, it is clear that the Pauli matrices anti-commute, that is, $\{X, Y\} = 0$, $\{X, Z\} = 0$, and $\{Y, Z\} = 0$ where $\{A, B\} = AB + BA$ is the anti-commutator. These Pauli matrices are used to describe the players' strategies. Eigenvectors of two-qubit operators created using these matrices and the single qubit identity matrix will be used as a basis for the players' measurements. The next table shows how to construct such an operator for each entry in the MS table so that they multiply in the row and columns to the identity or minus the identity. Observe that eigenvalues of these multiplications match the expected parity of the MS.

Another important observation is that the matrices in each column and row commute. As a consequence, they can be diagonalized in the same basis. This basis will be used

$\mathbb{I} \otimes Z$	$Z \otimes \mathbb{I}$	$Z \otimes Z$	$+\mathbb{I}$
$X \otimes \mathbb{I}$	$\mathbb{I} \otimes X$	$X \otimes X$	$+\mathbb{I}$
$X \otimes Z$	$Z \otimes X$	$Y \otimes Y$	$+\mathbb{I}$
$+\mathbb{I}$	$+\mathbb{I}$	$-\mathbb{I}$	

Table 3.3: Entangled players optimum strategy.

in the player's strategy. When Alice is asked about a row or column, suppose that she chooses an eigenvector $|\phi\rangle$ from that basis and send it to Bob. In this case, if Alice answers the referee with the eigenvalues of this eigenvector in her three operators, she will always be consistent with the required parity. Bob uses the operator in the entry asked for to determine the same thing, except that now for just one value. Bob's value is clearly equal to the value reported by Alice. Therefore, the players always win with this strategy. The problem is that Alice and Bob are not allowed to communicate once the game has started. One workaround is to split two EPR pairs between them, so that they share the state

$$\begin{aligned}
|\psi\rangle &= \left(\frac{|00\rangle^{AB} + |11\rangle^{AB}}{\sqrt{2}} \right) \otimes \left(\frac{|00\rangle^{AB} + |11\rangle^{AB}}{\sqrt{2}} \right) \\
&= \frac{1}{2} \sum_{i=0}^3 |i\rangle^A |i\rangle^B
\end{aligned}$$

which is a maximally entangled state. Since they are measuring in the same basis each half of the system, their answers will be completely correlated. For this reason, the communication that we mentioned can be replaced by this quantum state.

3.6.4 Tsirelson's characterization of **XOR** Games

Determining the value of a game when the players are allowed to share an arbitrary entangled state appears to be a difficult problem in the general case. The issue is that, a priori, there is no bound on the dimension of this shared state. Nevertheless, for two players **XOR** games, the Tsirelson Theorem provides a finite value for this dimension. It is stated as follows.

Theorem 3.6.5 (Tsirelson [95]). *Given an $n \times n$ complex matrix $C = (C)_{x,y}$, the following are equivalent:*

- (i) *there exist $d \in \mathbb{N}$, $|\psi\rangle \in \mathbb{C}^d \otimes \mathbb{C}^d$, $A_x, B_y \in \text{Herm}(\mathbb{C}^d)$, $A_x^2 = B_y^2 = \mathbb{I}$, such that $C_{x,y} = \langle \psi | A_x \otimes B_y | \psi \rangle$;*
- (ii) *there exist unit vectors $v_x, v_y \in \mathbb{R}^{n+2}$, for $1 \leq x, y \leq n$, such that $C_{x,y} = v_x v_y$;*
- (iii) *the same as (i) but $d \leq 2^{\lceil \frac{n+2}{2} \rceil}$.*

The second item in this theorem also guarantees that the value of the game can be approximated using Semi-definite Programming (SDP) [94]. Maximizing a constant function that depends on the inner products of unit vectors can be formulated as an SDP. Moreover, the size of this formulation is polynomial in the size of the game as these vectors live in \mathbb{R}^{n+2} . Since SDPs can be solved in polynomial time, the entangled value of a two player **XOR** game can be efficiently approximated within an exponentially small error, whereas approximating the classical value within a certain error is NP-hard. Nevertheless, more general quantum games can be NP-hard to approximate [61], or even QMA-hard to approximate [59].

3.7 Choi-Jamiolkowski Isomorphism

The quantum channel is a linear mapping $\Phi : \mathcal{Q} \rightarrow \mathcal{R}$ which is trace preserving and completely positive (CP). These two properties are necessary to ensure that input density operators are mapped to valid density operators. The first constraint guarantees that the trace remains one. While the second constraint guarantees that if Φ is applied to a subsystem A of a larger system on AR the resulting state is still positive. In a more mathematical notation, the operator $\Phi^A \otimes I^R$ maps the set of positive operators to itself for any reference system R . Quantum channels can be represented using the Kraus representation or the unitary equivalence acting on a environment [74]. Another convenient representation is through the isomorphism between quantum channels and quantum states known as Choi-Jamiolkowski isomorphism.

Entanglement is important in this representation as well. Let the space \mathcal{Q} be \mathbb{C}^n and \mathcal{R} be \mathbb{C}^m . Starting from the maximally entangled state $|\psi\rangle = \sum_i \frac{1}{\sqrt{n}} |i\rangle^A |i\rangle^B$ where $\dim(A) = \dim(B) = n$, the channel Φ is applied to the subsystem A as follows

$$(\Phi^A \otimes I^B) |\psi\rangle \langle \psi| = \sum_i \sum_j \frac{1}{n} \Phi(|i\rangle \langle j|^A) \otimes |i\rangle \langle j|^B.$$

The resulting state is the Choi-Jamiolkowski state corresponding to Φ . It may seem just an abstract representation of a channel, but it has practical applications in complexity [15].

3.8 Teleportation

Teleportation is the information processing task of remotely transferring the state of a quantum system from one location to another [16]. Once again entanglement is a main ingredient of this task. In the teleportation protocol, Alice and Bob share an EPR pair,

and Alice wants to teleport a state $|\psi\rangle = \alpha|0\rangle^{A_1} + \beta|1\rangle^{A_1}$. The EPR half under Alice's control lives on system A_2 , whereas Bob's lives in B . The joint state on Alice and Bob is

$$\begin{aligned} |\psi\rangle^{A_1} \otimes |\phi^+\rangle^{A_2B} &= \frac{\alpha}{\sqrt{2}}|0\rangle^{A_1}|00\rangle^{A_2B} + \frac{\alpha}{\sqrt{2}}|0\rangle^{A_1}|11\rangle^{A_2B} + \\ &\quad \frac{\beta}{\sqrt{2}}|1\rangle^{A_1}|00\rangle^{A_2B} + \frac{\beta}{\sqrt{2}}|1\rangle^{A_1}|11\rangle^{A_2B}. \end{aligned}$$

Alice measures her system A_1A_2 in the Bell basis. Rearranging their joint state on this basis, we have

$$\begin{aligned} &\frac{1}{2}[\alpha(|\phi^+\rangle + |\phi^-\rangle)^{A_1A_2}|0\rangle^B + \alpha(|\psi^+\rangle + |\psi^-\rangle)^{A_1A_2}|1\rangle^B + \\ &\quad \beta(|\psi^+\rangle - |\psi^-\rangle)^{A_1A_2}|0\rangle^B + \beta(|\phi^+\rangle - |\phi^-\rangle)^{A_1A_2}|1\rangle^B] \\ &= \frac{1}{2}[(|\phi^+\rangle^{A_1A_2}(\alpha|0\rangle + \beta|1\rangle)^B + |\phi^-\rangle^{A_1A_2}(\alpha|0\rangle - \beta|1\rangle)^B + \\ &\quad |\psi^+\rangle^{A_1A_2}(\alpha|1\rangle + \beta|0\rangle)^B + |\psi^-\rangle^{A_1A_2}(\alpha|1\rangle - \beta|0\rangle)^B]. \end{aligned}$$

No matter what Alice's outcome is in the Bell basis, Bob's system will have the amplitudes of the teleported qubit. Nonetheless, the resulting state on Bob's systems may need some adjustments to become equal to $|\psi\rangle$. There are four different cases that Alice must inform Bob so that he can apply the proper adjustments. For this, Alice sends Bob two classical bits informing the measured state on her system. The whole teleported process is not instantaneous and thus it does not violate any laws of relativity. Without Alice's message, Bob's EPR half will look like a totally mixed state to him. For each Bell state a classical two bit string is associated with it, which corresponds to the unitary Bob has to apply in his system. These correspondences are mapped as follows:

$$\begin{aligned} |\phi^+\rangle &\rightarrow |00\rangle \rightarrow I \\ |\phi^-\rangle &\rightarrow |01\rangle \rightarrow Z \\ |\psi^+\rangle &\rightarrow |10\rangle \rightarrow X \\ |\psi^-\rangle &\rightarrow |11\rangle \rightarrow ZX. \end{aligned}$$

The teleportation protocol is a proof of the resource inequality $[qq] + 2[c \rightarrow c] \geq [q \rightarrow q]$ where $[qq]$, $[c \rightarrow c]$, and $[q \rightarrow q]$ stand for one EPR pair, one access to a classical channel, and one access to a quantum channel, respectively. This means that having one EPR pair and making two accesses to a classical channel is at least as powerful in terms of resources as making one access to a quantum channel.

3.9 Superdense Coding

Quantum information can also improve resource usage when communicating classical information. In the superdense coding protocol, Alice can communicate two classical bits to Bob by using a single access to a quantum channel, given prior shared entanglement. A single shared EPR pair is enough to accomplish this. This entangled state is fixed and independent of the data being transmitted. Thus Alice and Bob can share as many EPR pairs as they want before the start of the protocol. A task that would require two channel access in the classical case can be done with a factor of two improvement. Evidently, this is not as interesting as an asymptotic gain. Despite living in an exponentially large space, a n -qubit state can not be used to encode asymptotically more than $O(n)$ bits to be decoded with constant success probability, in what is known as Holevo's bound [98]. This kind of information theoretical bound makes designing quantum communication protocols an ingenious art.

As we will show, the superdense protocol is very simple. Prior to its start, Alice and Bob share the EPR state $|\phi^+\rangle^{AB} = \frac{1}{\sqrt{2}}(|0\rangle^A|0\rangle^B + |1\rangle^A|1\rangle^B)$. As usual, Alice's state is on subsystem A while Bob is on B . We establish the following correspondance between two bit strings and Bell states:

$$\begin{aligned} |00\rangle &\leftrightarrow |\phi^+\rangle \\ |01\rangle &\leftrightarrow |\phi^-\rangle \\ |10\rangle &\leftrightarrow |\psi^+\rangle \\ |11\rangle &\leftrightarrow |\psi^-\rangle. \end{aligned}$$

Depending on the classical bit string Alice wants to communicate, she just needs to apply one of the operations $\{I, X, Z, ZX\}$ on her EPR half and send the resulting qubit to Bob. The application of these operators to $|\phi^+\rangle$ gives the following results

$$\begin{aligned} (I^A \otimes I^B)|\phi^+\rangle &= |\phi^+\rangle \\ (Z^A \otimes I^B)|\phi^+\rangle &= |\phi^-\rangle \\ (X^A \otimes I^B)|\phi^+\rangle &= |\psi^+\rangle \\ (ZX^A \otimes I^B)|\phi^+\rangle &= |\psi^-\rangle. \end{aligned}$$

Since the Bell states form an orthonormal basis for \mathbb{C}^4 , Bob can measure the received qubit with his own half to determine with certainty the classical two bit string. The superdense code protocol is also a proof of the resource inequality $[qq] + [q \rightarrow q] \geq 2[c \rightarrow c]$. Having one EPR pair and making one access to a quantum channel is at least as powerful as making two access to a classical channel.

Chapter 4

Quantum Tools

Quantum information is inherently more sophisticated than classical information. Thus, unique quantum tools had to be crafted to handle this more general form of information. Classical information can be trivially encoded in quantum states. A diagonal density matrix in the computational basis can be considered as containing only classical information. Moreover, when processing quantum information there is always the possibility of making a measurement in the computational basis to ensure that the state collapses to a definite classical state. Currently, we know how to accomplish a variety of tasks using quantum tools. However, the possibility of new ones seems far from exhausted. Their creation and any advances in existing ones may lead to major breakthroughs. In quantum complexity, protocols usually make extensive use of these tools. For instance, a tight analysis for the quantum de Finetti theorem in which the error is measured using the **SEP** norm may lead to the collapse $\text{QMA}(2) = \text{QMA}$.

Copying an arbitrary state is a simple task in the classical world, but it is not possible for arbitrary quantum states. Known tests for equality of two states also work differently in the quantum world. As an example, the swap test is only capable of comparing states that are not entangled and are close to pure. When states are orthogonal, this being the quantum way to say that they are completely different, this test still have a probability $\frac{1}{2}$ of failing by considering the states to be equal. Acquiring a classical description of a quantum state is an expensive task in terms of resources. It requires a number of independent copies that is exponential in its number of qubits. This process is known as quantum state tomography, and it has found applications in complexity theory despite being an important tool for experimentalists.

Comparing states using the swap test requires that they are not entangled. This is one of the reasons why bounding entanglement can be important. Quantum de Finetti theorems provide a way of understanding how much entanglement is present in permutation invariant states on $N + m$ subsystems when we are interested in the first m of them.

It provides a closeness guarantee to a separable state in some specific norm. For the trace norm, an appropriate value for N depends on the local dimension of the subsystems involved making this tool costly to use if this dimension is large. Contrary to quantum state tomography and some quantum de Finetti theorems, the quantum Fourier transform is an efficient, i.e. polynomial time, operation. It acts on the amplitudes of a quantum state instead of on an explicitly list of values as in the case of the discrete Fourier transform. It has applications in algorithms and complexity where it is used to detect periodicity, as in Shor's algorithm [91].

A generalization of the swap test, the product test, has interesting implications. As expected, it also needs two unentangled states, and it checks whether they are close to the same product state. This test allows us to show that $\text{QMA}(\text{poly}) = \text{QMA}(2)$.

In this section, we explore the tools mentioned so far. Note that there are important omissions, such as the tool for going from a verifier circuit in QMA to a set of local Hamiltonians. Having a good understanding of this process may shed light on the Quantum PCP Conjecture [6]. This chapter is a first endeavour to group tools that might be important to quantum complexity theory ¹.

4.1 No Cloning Theorem

As important as creating some quantum tools, it is to understand when they can not exist. A fundamental difference in the manipulation of quantum information with respect to the classical one is the inability to copy arbitrary unknown quantum states. This is a well know fact and goes by the name of “No-cloning Theorem”. A facet of this limitation was captured by Scott Aaronson when he mentioned its relation with the Quantum PCP: “I’m quite certain that a Quantum PCP Theorem will require significant new ideas. Recently I spent a day or two studying Irit’s proof of the classical PCP theorem (which I hadn’t done before), and I found about 20 violations of the No-cloning Theorem on every page.” [2].

Theorem 4.1.1 (No-cloning Theorem). *There is no unitary U such that for all states $|\psi\rangle$ and a fixed state $|e\rangle$, it holds that $U|\psi\rangle|e\rangle = |\psi\rangle|\psi\rangle$.*

Proof. Let $|\psi\rangle$ and $|\phi\rangle$ be two arbitrary states. The inner product $|\psi\rangle$ with $|\phi\rangle$ is

$$\begin{aligned} \langle\psi|\phi\rangle &= \langle\psi|^A \langle e|^B |\phi\rangle^A |e\rangle^B \\ &= \langle\psi|^A \langle e|^B U^\dagger U |\phi\rangle^A |e\rangle^B \\ &= \langle\psi|^A \langle\psi|^B |\phi\rangle^A |\phi\rangle^B \\ &= \langle\psi|\phi\rangle^2. \end{aligned}$$

¹For any suggestions, please contact the author via the e-mail: fegranha@gmail.com

The previous equality only holds when $\langle \psi | \phi \rangle$ is equal to 0 or to 1. For arbitrary states, it does not hold. \square

4.2 Quantum State Tomography

After creating a quantum operation in a laboratory, an experimental physicist may want to fully determine states before and after the application of this new operation, to make sure the operation was correctly implemented. This process of acquiring a classical representation of a state is known as quantum state tomography, and it requires a certain number of independent copies of the state. The description of a n -qubit state ρ is itself exponentially large in n , and so it is natural that quantum state tomography might require an exponential number of copies. Tomography is not only confined to the experimental realm, it was proven useful in the context of computational complexity to ensure that a state received from a prover has a certain form, as pointed out in [15]. The implementation of this process may vary depending on the measurements and operators used. In this presentation we follow the implementation details given in [15].

The copies of the unknown state ρ are measured using an information complete POVM $\{P_a\}_{a \in \Gamma}$ to estimate the probabilities

$$p(a) = \langle P_a, \rho \rangle,$$

where $\Gamma = \{0, 1, 2, 3\}$ for a 2×2 density matrix. Associated with these measurements there is a set of operators $\{M_a\}_{a \in \Gamma}$ which allows the reconstruction of any density operator X as

$$\sum_{a \in \Gamma} M_a \langle P_a, X \rangle = X.$$

In practice, the probabilities $p(a)$ can only be estimated. We denote by $q(a)$ the empirical mean associated with outcome a , that is, $q(a) = \frac{\# \text{ of outcomes } a}{N}$ where N is the total number of measured copies. The error in the trace norm of the reconstructed state from the estimates is bounded by

$$\left\| \sum_{a \in \Gamma} p(a) M_a - \sum_{a \in \Gamma} q(a) M_a \right\|_1 \leq \sum_{a \in \Gamma} |p(a) - q(a)| \|M_a\|_1 \leq \|p - q\|_1 \max_{a \in \Gamma} \|M_a\|_1.$$

A particular set of measurements that can be implemented exactly using Hadamard gates, controlled not gates, $\frac{\pi}{8}$ -phase shift gates, and measurements in the computational basis is

$$P_0 = \begin{pmatrix} \frac{2+\sqrt{2}}{8} & \frac{1+i}{8} \\ \frac{1-i}{8} & \frac{2-\sqrt{2}}{8} \end{pmatrix}, P_1 = \begin{pmatrix} \frac{2-\sqrt{2}}{8} & \frac{1-i}{8} \\ \frac{1+i}{8} & \frac{2+\sqrt{2}}{8} \end{pmatrix},$$

$$P_2 = \begin{pmatrix} \frac{2+\sqrt{2}}{8} & \frac{-1-i}{8} \\ \frac{-1+i}{8} & \frac{2-\sqrt{2}}{8} \end{pmatrix}, P_3 = \begin{pmatrix} \frac{2-\sqrt{2}}{8} & \frac{-1+i}{8} \\ \frac{-1-i}{8} & \frac{2+\sqrt{2}}{8} \end{pmatrix}.$$

Their associated operators are

$$M_0 = \begin{pmatrix} \frac{1+\sqrt{2}}{2} & \frac{1+i}{2} \\ \frac{1-i}{2} & \frac{1-\sqrt{2}}{2} \end{pmatrix}, M_1 = \begin{pmatrix} \frac{1-\sqrt{2}}{2} & \frac{1-i}{2} \\ \frac{1+i}{2} & \frac{1+\sqrt{2}}{2} \end{pmatrix},$$

$$M_2 = \begin{pmatrix} \frac{1+\sqrt{2}}{2} & \frac{-1-i}{2} \\ \frac{-1+i}{2} & \frac{1-\sqrt{2}}{2} \end{pmatrix}, M_3 = \begin{pmatrix} \frac{1-\sqrt{2}}{2} & \frac{-1+i}{2} \\ \frac{-1-i}{2} & \frac{1+\sqrt{2}}{2} \end{pmatrix},$$

where $\|M_a\|_1 < 4$, for $a \in \Gamma = \{0, 1, 2, 3\}$.

Generalizing the tomography to k -qubits states results in the following measurements

$$P_x = P_{x_1} \otimes \cdots \otimes P_{x_k},$$

and operators

$$M_x = M_{x_1} \otimes \cdots \otimes M_{x_k},$$

where $x = x_1 \dots x_k \in \Gamma^k$. Note that the norm is bounded by $\|M_x\|_1 < 4^k$ since the operators are built using tensor products of matrices with $\|M_a\| < 4$.

We denote the reconstructed state using the approximate probability distribution q as H , and it is given by

$$H = \sum_{x \in \Gamma^k} q(x) M_x.$$

Lemma 4.2.1 (From [15]). *For any choice of $\epsilon > 0$, taking $N \geq \frac{2^{10k}}{\epsilon^3}$ will guarantee that with probability at least $1 - \epsilon$ the estimate H satisfies $\|\rho - H\|_1 < \epsilon$.*

Proof. A crucial point in the analysis of the quantum state tomography is acquiring good estimates of q with high probability. The probability of the empirical mean deviating from the actual mean for a single $x \in \Gamma^k$ can be bounded by the Hoeffding's inequality as

$$\Pr[|q(x) - p(x)| \geq \delta] \leq 2 \exp(-2N\delta^2).$$

The deviation of the whole probability vector q from p can be bounded as

$$\begin{aligned} \Pr[\|q - p\|_1 \geq 4^k \delta] &\leq \Pr[|q(x) - p(x)| \geq \delta \text{ for at least one } x \in \Gamma^k] \\ &\leq \Pr[\cup_{x \in \Gamma^k} |q(x) - p(x)| \geq \delta], \end{aligned}$$

in which the last inequality is due to the union bound. Applying the Hoeffding's inequality, we have

$$\Pr[\|q - p\|_1 \geq 4^k \delta] \leq 2^{2k+1} \exp(-2N\delta^2).$$

Taking $\delta = \frac{\epsilon}{16^k}$ and using the hypothesis of $N \geq \frac{2^{10k}}{\epsilon^3}$, the argument inside the exponential becomes

$$-2N\delta^2 = -\frac{2N\epsilon^2}{2^{8k}} \leq -\frac{2^{2k+1}}{\epsilon}.$$

Replacing the previous value in the probability expression results in

$$\Pr[\|q - p\|_1 \geq \frac{\epsilon}{4^k}] \leq 2^{2k+1} \exp(-\frac{2^{2k+1}}{\epsilon}) < \epsilon,$$

where the last inequality follows from the fact that $e^{-\alpha} < \frac{1}{\alpha}$, for all $\alpha > 0$. The probability that the reconstructed state differs from the actual state is

$$\begin{aligned} \Pr[\|\rho - H\|_1 \geq \epsilon] &\leq \Pr[\|p - q\|_1 \max_{x \in \Gamma^k} \|M_x\|_1 \geq \epsilon] \\ &= \Pr[\|p - q\|_1 \geq \frac{\epsilon}{4^k}] < \epsilon. \end{aligned}$$

□

4.3 Quantum de Finetti

The study of the de Finetti theorems initiated with the investigation of properties of probability distributions. Given a probability distribution $p(x_1, \dots, x_n)$ on n variables, if this distribution remains the same for every possible permutation of these n variables, then it is denoted permutation invariant or exchangeable. Suppose p is exchangeable. Let $p_k(x_1, \dots, x_k)$ be the marginal of p on the first $k \leq n$ variables. These de Finetti theorems provide bounds on the distance between p_k and a convex combination of product distributions on k variables of the form $\int r(x_1) \dots r(x_k) \mu(r)$. Let d denote the possible values of a variable x_i . Two possible bounds are

$$\left\| p_k(x_1, \dots, x_k) - \int r(x_1) \dots r(x_k) \mu(r) \right\|_1 \leq \frac{2kd}{n},$$

and

$$\left\| p_k(x_1, \dots, x_k) - \int r(x_1) \dots r(x_k) \mu(r) \right\|_1 \leq \frac{k(k-1)}{n},$$

from [28]. Note that the second bound is independent of the “local dimension” of each variable. Finite quantum de Finetti theorems are similar in spirit, but the objects are quantum states of the form $\rho^{A_1 \dots A_n}$ instead of probability distributions. They bound the distance of $\rho^{A_1 \dots A_k}$ from a convex combination of independent and identically distributed (i.i.d.) states of the form $\int \xi^{\otimes k} \mu(\xi)$ provided that the state $\rho^{A_1 \dots A_n}$ is permutation invariant or exchangeable, *i.e.* remains invariant under any permutation of subsystems A_1 through A_n :

$$\left\| \rho^{A_1 \dots A_k} - \int \xi^{\otimes k} \mu(\xi) \right\|_1 \leq 4 \frac{d^2 k}{n}.$$

This convex combination of i.i.d. states is a separable state. Therefore, the quantum de Finetti theorems are useful to limit the amount of entanglement across the partitioned subsystems. Similarly to the classical case, as n grows relatively to k , the bound decreases. This kind of theorem is a proof that entanglement is monogamous, meaning that a quantum system can not be highly entangled with many others at the same time.

The quantum de Finetti theorem for the trace norm we present relies on the properties of the symmetric subspace. When compared to the dimension of the whole space, the dimension of this space becomes small for n larger than d . This fact limits the possible entanglement correlations among subsystems. Contrary to the classical case, the quantum bound depends on the local dimension. There is one important caveat in applying the quantum de Finetti for the trace norm. For instance, when the local system is composed of polynomially many qubits, the number of subsystems n needs to be exponential large to get a constant error bound. For this reason, de Finetti theorems for restricted norms were created with a logarithmic dependency on the local dimension, making them less costly to use.

4.3.1 The Symmetric Subspace

The Symmetric Subspace has important applications in quantum information, besides de Finetti theorems [50]. In this section, we formally defined it and present some results. This presentation is based on the work of Harrow [50] and Watrous [97].

Let S_n be the symmetric group on n elements. For each permutation π , we associate the operator

$$P_d(\pi) = \sum_{i_1, \dots, i_n \in [d]} |i_{\pi^{-1}(1)}, \dots, i_{\pi^{-1}(n)}\rangle \langle i_1, \dots, i_n|.$$

With this representation the composition of permutations π_1 and π_2 is simply $P_d(\pi_1 \pi_2) = P_d(\pi_1) P_d(\pi_2)$. This means that the group multiplication operation can be done by matrix multiplication characterizing a representation of S_n in $(\mathbb{C}^d)^{\otimes n}$.

The *symmetric subspace* denoted by $\vee^n \mathbb{C}^d$ is the set of all vectors in $(\mathbb{C}^d)^{\otimes n}$ that are invariant under all permutations in S_n , that is

$$\vee^n \mathbb{C}^d = \{|\psi\rangle : P_d(\pi)|\psi\rangle = |\psi\rangle \text{ for all } \pi \in S_n\}.$$

The symmetric subspace has the following orthogonal projector

$$P_{sym}^{d,n} = \frac{1}{n!} \sum_{\pi \in S_n} P_d(\pi).$$

Lemma 4.3.1. $P_{sym}^{d,n}$ is the orthogonal projector onto $\vee^n \mathbb{C}^d$.

Proof. A necessary and sufficient condition for an operator Π to be an orthogonal projector is $\Pi^\dagger \Pi = \Pi$, sufficing to show that $(P_{sym}^{d,n})^\dagger P_{sym}^{d,n} = P_{sym}^{d,n}$.

Since $P_{sym}^{d,n}$ is an equal combination of permutation representations, multiplying it by any permutation π does not change it. Indeed,

$$\begin{aligned} P_d(\pi) P_{sym}^{d,n} &= P_d(\pi) \frac{1}{n!} \sum_{\pi' \in S_n} P_d(\pi') \\ &= \frac{1}{n!} \sum_{\pi' \in S_n} P_d(\pi \pi') \\ &= \frac{1}{n!} \sum_{\pi^{-1} \pi'' \in S_n} P_d(\pi'') \\ &= \frac{1}{n!} \sum_{\pi'' \in S_n} P_d(\pi'') = P_{sym}^{d,n}. \end{aligned}$$

This invariance implies that $P_{sym}^{d,n}$ satisfies the orthogonal projector criteria

$$(P_{sym}^{d,n})^\dagger P_{sym}^{d,n} = \frac{1}{n!} \sum_{\pi \in S_n} P_d(\pi^{-1}) P_{sym}^{d,n} = \frac{1}{n!} \sum_{\pi \in S_n} P_{sym}^{d,n} = P_{sym}^{d,n}.$$

It remains to show that $P_{sym}^{d,n}$ is a projector onto $\vee^n \mathbb{C}^d$. For any permutation π , a projected vector $P_{sym}^{d,n}|\psi\rangle$ satisfies

$$P_d(\pi) P_{sym}^{d,n}|\psi\rangle = P_{sym}^{d,n}|\psi\rangle.$$

This implies that the image of the operator $\text{Im} P_{sym}^{d,n}$ is contained in the symmetric subspace. The containment $\vee^n \mathbb{C}^d \subseteq \text{Im} P_{sym}^{d,n}$ also holds since for any $|\psi\rangle \in \vee^n \mathbb{C}^d$, we have

$$P_{sym}^{d,n}|\psi\rangle = \frac{1}{n!} \sum_{\pi \in S_n} P_d(\pi)|\psi\rangle = |\psi\rangle.$$

□

There are two alternative characterizations of $\mathbb{V}^n(\mathbb{C}^d)$. The first characterization consists in the span of all product vectors, defined as

$$A = \text{span}\{|\Phi\rangle^{\otimes n} : \Phi \in \mathbb{C}^d\}.$$

Let $\vec{i} = (i_1, \dots, i_n)$ be a vector in $[d]^n$. The operation $T(\vec{i})$ associates a d -tuple $\vec{t} = (t_1, \dots, t_d)$, denoted *type* of \vec{i} , in which t_j counts the number of occurrences of the value j in \vec{i} . For a given n , the set $\mathbb{I}_{d,n}$ contains all \vec{t} consistent with \vec{i} of size n , that is

$$\mathbb{I}_{d,n} = \{\vec{t} = (t_1, \dots, t_d) : t_1 + \dots + t_d = n\}.$$

For a given type \vec{t} the number of possibilities for \vec{i} is

$$|T^{-1}(\vec{t})| = \frac{n!}{t_1! \dots t_d!} = \binom{n}{\vec{t}}.$$

Moreover, for each \vec{t} we can associate the vector

$$|s_{\vec{t}}\rangle = \binom{n}{\vec{t}}^{-\frac{1}{2}} \sum_{\vec{i}: T(\vec{i})=\vec{t}} |i_1, \dots, i_n\rangle.$$

These vectors form an orthonormal basis for $\mathbb{V}^n(\mathbb{C}^d)$. The fact that they are orthonormal can be readily verified. The number of base vectors is equal to the number of different *types* $\vec{t} = (t_1, \dots, t_d)$. Since they must satisfy $t_1 + \dots + t_d = n$ and each t_j can be zero, the total number of them is $\binom{n+d-1}{d-1}$. When d is small compared to n , the dimension of the symmetric space becomes much smaller than the dimension of the whole space $(\mathbb{C}^d)^{\otimes n}$. This is perhaps the most important characteristic of $\mathbb{V}^n(\mathbb{C}^d)$ for quantum information. Projecting a quantum state onto $\mathbb{V}^n(\mathbb{C}^d)$ can greatly reduce the freedom of the state and consequently reduce the entanglement.

When the local dimension d is 2 and n is 3, there are 4 base vectors corresponding to the types $\vec{t}_1 = (3, 0)$, $\vec{t}_2 = (0, 3)$, $\vec{t}_3 = (2, 1)$, and $\vec{t}_4 = (1, 2)$ as illustrated next:

$$\begin{aligned} |s_{\vec{t}_1}\rangle &= |1, 1, 1\rangle \\ |s_{\vec{t}_2}\rangle &= |2, 2, 2\rangle \\ |s_{\vec{t}_3}\rangle &= \frac{1}{\sqrt{3}}(|1, 1, 2\rangle + |1, 2, 1\rangle + |2, 1, 1\rangle) \\ |s_{\vec{t}_4}\rangle &= \frac{1}{\sqrt{3}}(|1, 2, 2\rangle + |2, 1, 2\rangle + |2, 2, 1\rangle). \end{aligned}$$

An equivalent definition of A is

$$B = \text{span}\{|s_{\vec{t}}\rangle : \vec{t} \in \mathbb{I}_{d,n}\}.$$

This fact is shown in the Lemma 4.3.3. Before going through the details of its proof, a lemma about polynomials is needed. It states that if the values of a polynomial belong to a certain subspace W for all inputs, then this polynomial has coefficients in W .

Lemma 4.3.2. *Let $p(x_1, \dots, x_d) = \sum_{\vec{t} \in [d]^n} c_{\vec{t}} x_1^{t_1} \dots x_d^{t_d}$, be a polynomial with coefficients in a finite dimensional space V . If for all values of x_1, \dots, x_d the value $p(x_1, \dots, x_d)$ belongs to W for $W \subset V$, then $c_{\vec{t}} \in W$.*

Lemma 4.3.3. $\vee^n(\mathbb{C}^d) = A = B$.

Proof. From the definition of A and B , it is easy to see that $A \subseteq \vee^n(\mathbb{C}^d)$ and also $B \subseteq \vee^n(\mathbb{C}^d)$.

The containment $\vee^n(\mathbb{C}^d) \subseteq B$ also follows easily. Let $|i_1, \dots, i_n\rangle$ be a given vector and let \vec{t} be its *type*. Note that

$$P_{sym}^{d,n} |i_1, \dots, i_n\rangle = \binom{n}{\vec{t}}^{-1} \sum_{\vec{i}: T(\vec{i})=\vec{t}} |i_1, \dots, i_n\rangle = \binom{n}{\vec{t}}^{-\frac{1}{2}} |s_{\vec{t}}\rangle.$$

This means that $\text{Im } P_{sym}^{d,n} \subseteq B$. As it also holds that $\text{Im } P_{sym}^{d,n} = \vee^n(\mathbb{C}^d)$, the containment $\vee^n(\mathbb{C}^d) \subseteq B$ follows.

Finally, we show that $B \subseteq A$ by considering the state $|p(x_1, \dots, x_d)\rangle = (\sum_{i=1}^d x_i |i\rangle)^{\otimes n}$ that is certainly in A . Expanding this state, we get

$$|p(x_1, \dots, x_d)\rangle = \sum_{\vec{t} \in \mathbb{I}_{d,n}} x_1^{t_1} \dots x_d^{t_d} \binom{n}{\vec{t}} \sum_{\vec{i}: T(\vec{i})=\vec{t}} |i_1, \dots, i_n\rangle.$$

This implies that the coefficients of $x_1^{t_1} \dots x_d^{t_d}$ are proportional to $|s_{\vec{t}}\rangle$. Since the state $|p(x_1, \dots, x_d)\rangle$ belongs to A no matter the values of x_1, \dots, x_d , Lemma 4.3.2 guarantees that the coefficients also belong to A . Therefore, all $|s_{\vec{t}}\rangle$ belong to A , for all $\vec{t} \in \mathbb{I}_{d,n}$. As the span of these vectors is equal to B , we have $B \subseteq A$, concluding the proof. \square

A generalization of orthogonal projectors may be given for a group G . Suppose that $f : G \rightarrow \mathbb{C}$ is an integrable function and μ is a measure. We say that μ is an invariant measure if for all $g \in G$ it holds that $\int_{x \in G} f(x) \mu(x) dx = \int_{x \in G} f(gx) \mu(x) dx$. A finite group always has the invariant measure $\frac{1}{|G|}$. In the case of the unitary group, there is a unique invariant measure known as the Haar measure.

Lemma 4.3.4. *Let G be a group with an invariant measure μ , and a representation $R : G \rightarrow L(V)$ where $L(V)$ is the set of linear operators from space V to itself. If it holds that*

$$V^G = \{|\psi\rangle \in V : R(g)|\psi\rangle = |\psi\rangle \ \forall g \in G\} \text{ and}$$

$$\Pi = \int_{x \in G} R(x) \mu(x) dx,$$

then Π is an orthogonal projector onto V^G .

Proof. The proof follows the same steps when we proved that $P_{sym}^{d,n}$ is an orthogonal projector. We first note that Π is unaffected by a multiplication of $R(g)$, for any g . This is only true because μ is an invariant measure.

$$R(g)\Pi = \int_{x \in G} R(g)R(x)\mu(x)dx = \int_{x \in G} R(gx)\mu(x)dx = \int_{x \in G} R(gx)\mu(x)dx = \Pi$$

Now, the necessary and sufficient property for orthogonal operators follows:

$$\Pi^\dagger \Pi = \int_{x \in G} R(x^{-1})\mu(x)dx \Pi = \int_{x \in G} \Pi \mu(x)dx = \Pi \int_{x \in G} \mu(x)dx = \Pi.$$

□

The projector onto the symmetric subspace $\vee^m(\mathbb{C}^d)$ can be also written as an integral in the form

$$S^{(m)} = \binom{m+d-1}{d-1} \int |\psi\rangle \langle \psi|^{\otimes m} d\mu(|\psi\rangle),$$

where $|\psi\rangle$ is an unit vector in \mathbb{C}^d .

4.3.2 Quantum de Finetti Theorem for the Trace Norm

Dealing with a pure state may be simpler than dealing with a density operator. The following lemma states that any exchangeable operator admits a purification that is in the symmetric subspace.

Lemma 4.3.5 (Adapted from [97]). *Let ρ^{A_1, \dots, A_n} be a quantum state on registers A_1 through A_n , where each register is in the space \mathbb{C}^d . If ρ^{A_1, \dots, A_n} is exchangeable, then there is a purification $|\psi\rangle^{A'_1, \dots, A'_n} \in \vee^n(\mathbb{C}^{d^2})$ on registers A'_1 through A'_n , where each register is in space \mathbb{C}^{d^2} .*

We state and prove a quantum de Finetti Theorem for the general trace norm.

Theorem 4.3.6 (Adapted from [97]). *Let A_1, \dots, A_n be identical quantum registers, each having an associated space \mathbb{C}^d , and let $\rho \in D(A_1 \otimes \dots \otimes A_n)$ be an exchangeable density operator representing the state of these registers. For any choice of $k \in \{1, \dots, n\}$, there exists a finite set Γ , a probability vector $p \in \mathbb{R}^\Gamma$, and a collection of density operators $\{\xi_a : a \in \Gamma\} \subseteq D(\mathbb{C}^d)$ such that*

$$\left\| \rho^{A_1 \dots A_k} - \sum_{a \in \Gamma} p(a) \xi_a^{\otimes k} \right\|_1 < \frac{4d^2 k}{n}.$$

Proof. Using Lemma 4.3.5, it is enough to restrict our attention to pure states. We derive the stronger upper bound of

$$\frac{4dk}{n}$$

that can be readily adapted for exchangeable density matrices by changing the local dimension from d to d^2 .

Let $|\psi\rangle^{A_1, \dots, A_n}$ be a pure state in $\mathcal{V}^n(\mathbb{C}^d)$. For clarity we suppress the superscript A_1, \dots, A_n thereafter. We denote the space of the first k registers by $\mathcal{Y} = \mathbb{C}^{\otimes k}$ and the space of the remaining registers by $\mathcal{Z} = \mathbb{C}^{\otimes(n-k)}$. Since $|\psi\rangle$ belongs to the symmetric subspace, projecting any number of registers to their corresponding symmetric subspace does not change $|\psi\rangle$. More specifically, this holds when the last $n - k$ registers are projected:

$$(\mathbb{I}_{\mathcal{Y}} \otimes S^{(n-k)})|\psi\rangle = |\psi\rangle.$$

We are interested in an approximation of $\sigma = \text{Tr}_{\mathcal{Z}}(|\psi\rangle\langle\psi|)$ by a separable state. Note that this state σ can be equivalently written as

$$\sigma = \text{Tr}_{\mathcal{Z}}((\mathbb{I}_{\mathcal{Y}} \otimes S^{(n-k)})|\psi\rangle\langle\psi|).$$

We introduce an operator to simplify the notation when $S^{(n-k)}$ is applied to a density matrix. For each vector $|\phi\rangle$, we define the operator $\Phi_{|\phi\rangle} : \mathcal{Y} \otimes \mathcal{Z} \rightarrow \mathcal{Y}$ as

$$\Phi_{|\phi\rangle}(X) = (\mathbb{I}_{\mathcal{Y}} \otimes |\phi\rangle\langle\phi|^{\otimes(n-k)})X(\mathbb{I}_{\mathcal{Y}} \otimes |\phi\rangle\langle\phi|^{\otimes(n-k)})^\dagger.$$

In terms of this new operator $\Phi_{|\phi\rangle}$, the state σ becomes

$$\sigma = \binom{n-k+d-1}{d-1} \int \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) d\mu(|\phi\rangle).$$

We are looking for a separable state τ that is sufficiently close to σ in order to attain the claimed distance bound. A tentative state τ is

$$\tau = \binom{n+d-1}{d-1} \int \langle(|\phi\rangle\langle\phi|)^{\otimes k}, \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|)\rangle (|\phi\rangle\langle\phi|)^{\otimes k} d\mu(|\phi\rangle).$$

Now, we focus on showing that τ indeed attains the desired bound. Observe that it belongs to the convex hull of separable states.

$$\tau \in \text{conv}\{(|\phi\rangle\langle\phi|)^{\otimes k} | |\phi\rangle \in \mathbb{C}^d\}$$

To alleviate the notation once more, we introduce the term c_m as

$$c_m = \binom{m+d-1}{d-1}.$$

This term is simply the dimension of the space $\vee^m(\mathbb{C}^d)$.

Now, we proceed to estimate the distance $\|\sigma - \tau\|_1$ by deriving the bound

$$\begin{aligned} \|\sigma - \tau\|_1 &\leq \left\| \sigma - \frac{c_{n-k}}{c_n} \tau \right\|_1 + \left\| \frac{c_{n-k}}{c_n} \tau - \tau \right\|_1 \\ &= c_{n-k} \left\| \frac{1}{c_{n-k}} \sigma - \frac{1}{c_n} \tau \right\|_1 + \left(1 - \frac{c_{n-k}}{c_n}\right), \end{aligned}$$

where the first inequality follows from the triangle inequality.

To bound the term $\left\| \frac{1}{c_{n-k}} \sigma - \frac{1}{c_n} \tau \right\|_1$, we use a simple fact about operators A and B :

$$A - BAB = A(\mathbb{I} - B) + (\mathbb{I} - B)A - (\mathbb{I} - B)A(\mathbb{I} - B).$$

We introduce another notation simplification by letting $\Delta_\phi = (|\phi\rangle\langle\phi|)^{\otimes k}$. Using this notation and the previous fact about operators, we have

$$\begin{aligned} \left\| \frac{1}{c_{n-k}} \sigma - \frac{1}{c_n} \tau \right\|_1 &= \left\| \int (\Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) - \Delta_\phi \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) \Delta_\phi) d\mu(|\phi\rangle) \right\|_1 \\ &\leq \left\| \int \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) (\mathbb{I}_Y - \Delta_\phi) d\mu(|\phi\rangle) \right\|_1 + \left\| \int (\mathbb{I}_Y - \Delta_\phi) \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) d\mu(|\phi\rangle) \right\|_1 \\ &\quad \left\| \int (\mathbb{I}_Y - \Delta_\phi) \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) (\mathbb{I}_Y - \Delta_\phi) d\mu(|\phi\rangle) \right\|_1. \end{aligned}$$

Due to the cyclic property of the trace, the following equality holds

$$\left\| \int \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) (\mathbb{I}_Y - \Delta_\phi) d\mu(|\phi\rangle) \right\|_1 = \left\| \int (\mathbb{I}_Y - \Delta_\phi) \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) d\mu(|\phi\rangle) \right\|_1.$$

The term $(\mathbb{I}_Y - \Delta_\phi)$ is simply an orthogonal projector, and the following inequality holds:

$$\left\| \int (\mathbb{I}_Y - \Delta_\phi) \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) (\mathbb{I}_Y - \Delta_\phi) d\mu(|\phi\rangle) \right\|_1 \leq \left\| \int (\mathbb{I}_Y - \Delta_\phi) \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) d\mu(|\phi\rangle) \right\|_1.$$

The upper bound on $\left\| \frac{1}{c_{n-k}} \sigma - \frac{1}{c_n} \tau \right\|_1$ can be written as

$$\left\| \frac{1}{c_{n-k}} \sigma - \frac{1}{c_n} \tau \right\|_1 \leq 3 \left\| \int (\mathbb{I}_Y - \Delta_\phi) \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) d\mu(|\phi\rangle) \right\|_1.$$

Remember that $|\psi\rangle$ is invariant under projections to the symmetric subspace, for any number of registers. This fact leads to

$$\int \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) d\mu(|\phi\rangle) = \frac{1}{c_{n-k}} \sigma$$

and so

$$\begin{aligned} \int \Delta_\phi \Phi_{|\phi\rangle}(|\psi\rangle\langle\psi|) d\mu(|\phi\rangle) &= \text{Tr}_{\mathcal{Z}} \left(\int (|\phi\rangle\langle\phi|)^{\otimes n} |\psi\rangle\langle\psi| d\mu(|\phi\rangle) \right) \\ &= \frac{1}{c_n} \sigma. \end{aligned}$$

We have the explicit bound

$$\left\| \frac{1}{c_{n-k}} \sigma - \frac{1}{c_n} \tau \right\|_1 \leq 3 \left(\frac{1}{c_{n-k}} - \frac{1}{c_n} \right).$$

Using this bound, in the original distance $\|\sigma - \tau\|_1$ results in

$$\|\sigma - \tau\|_1 \leq 3c_{n-k} \left(\frac{1}{c_{n-k}} - \frac{1}{c_n} \right) + \left(1 - \frac{c_{n-k}}{c_n} \right) = 4 \left(1 - \frac{c_{n-k}}{c_n} \right).$$

We compute a bound to the ratio $\frac{c_{n-k}}{c_n}$ as

$$\begin{aligned} \frac{c_{n-k}}{c_n} &= \frac{(n-k+d-1)(n-k+d-2) \cdots (n-k+1)}{(n+d-1)(n+d-2) \cdots (n+1)} \\ &\leq \left(\frac{n-k+1}{n+1} \right)^{d-1} \\ &< 1 - \frac{dk}{n}. \end{aligned}$$

The stronger bound for pure states

$$\|\sigma - \tau\|_1 < \frac{4dk}{n},$$

is established. For general exchangeable operators, we use the Lemma 4.3.5 as already explained, to derive a final bound of

$$\|\sigma - \tau\|_1 < \frac{4d^2k}{n}.$$

□

4.4 Operation from State

The Choi-Jamiołkowski (CJ) representations is not just a mere conceptual tool. Beige et al. [15] showed how to use a CJ state to simulate the application of the corresponding operator in an arbitrary mixed state. They achieve this result by measuring both states in a certain basis, and post-selecting according to the outcome. Let

$|\psi\rangle\langle\psi| = \sum_i \sum_j \frac{1}{2^n} \Phi(|i\rangle\langle j|^A) \otimes |i\rangle\langle j|^B$ be the CJ representation of $\Phi : \mathcal{Q} \rightarrow \mathcal{R}$ and ξ^C be an arbitrary mixed state in a space isomorphic to \mathcal{Q} . Note that ξ^C can possibly be entangled with another system. Denote by n the number of qubits in the input space. From 1 to n , each qubit in system B , and the corresponding one in C , are measured in the Bell basis. The simulation accepts if and only if all outcomes are $|\phi^+\rangle$, in which case we have

$$\begin{aligned} & \sum_{i'} \sum_{j'} \frac{1}{4^n} \sum_i \sum_j \Phi(|i\rangle\langle j|^A) \langle i'|i\rangle \langle j|j'\rangle \langle i'|\xi^C|j'\rangle \\ &= \frac{1}{4^n} \sum_i \sum_j \Phi(|i\rangle\langle j|^A) \langle i|\xi^C|j\rangle = \frac{1}{4^n} \Phi(\sum_i \sum_j \langle i|\xi^C|j\rangle |i\rangle\langle j|^A) = \frac{1}{4^n} \Phi(\xi^C). \end{aligned}$$

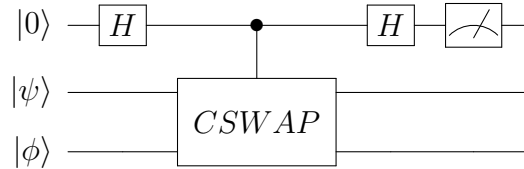
The probability of measuring a single $|\phi^+\rangle$ is $\frac{1}{4}$, thus this procedure has a $\frac{1}{4^n}$ success probability. For n of logarithmic size this procedure is still efficient.

4.5 Swap Test

The swap test was designed to check if two unentangled states $|\psi\rangle$ and $|\phi\rangle$ are close. The acceptance probability of this test depends on the overlap $|\langle\psi|\phi\rangle|^2$. When the states are the same, the acceptance probability is 1, and it is $\frac{1}{2}$ for orthogonal states. The implementation of this test uses the controlled swap gate (CSWAP) whose action on basis states $|i\rangle$ and $|j\rangle$ can be described as

$$\begin{aligned} |0\rangle|i\rangle|j\rangle &\rightarrow |0\rangle|i\rangle|j\rangle \\ |1\rangle|i\rangle|j\rangle &\rightarrow |1\rangle|j\rangle|i\rangle. \end{aligned}$$

Before applying the CSWAP gate, the control qubit is put in a uniform superposition by the application of the Hadamard gate. This allows the CSWAP to act on half of the superposition $\frac{1}{\sqrt{2}}(|0\rangle|\psi\rangle|\phi\rangle + |1\rangle|\phi\rangle|\psi\rangle)$. If the input states are equal, the first control qubit remains unentangled with the input states, and a further application of the Hadamard gate takes it back to $|0\rangle$. For this reason, we associate measuring $|0\rangle$ in the control qubit as the acceptance condition. The swap test circuit is depicted next.



Now, we investigate the effect of the circuit more generally on the initial state $|0\rangle|\psi\rangle|\phi\rangle$, we get

$$\begin{aligned}
(H \otimes I)\text{CSWAP}(H \otimes I)|0\rangle|\psi\rangle|\phi\rangle &= (H \otimes I)\text{CSWAP}\left(\frac{1}{\sqrt{2}}|0\rangle|\psi\rangle|\phi\rangle + \frac{1}{\sqrt{2}}|1\rangle|\psi\rangle|\phi\rangle\right) \\
&= (H \otimes I)\left(\frac{1}{\sqrt{2}}|0\rangle|\psi\rangle|\phi\rangle + \frac{1}{\sqrt{2}}|1\rangle|\phi\rangle|\psi\rangle\right) \\
&= \frac{1}{2}|0\rangle(|\psi\rangle|\phi\rangle + |\phi\rangle|\psi\rangle) + \frac{1}{2}|1\rangle(|\psi\rangle|\phi\rangle - |\phi\rangle|\psi\rangle).
\end{aligned}$$

To get a hold on the acceptance probability, the state $|\psi\rangle$ is written in terms of $|\phi\rangle$ and $|\phi^\perp\rangle$ as $|\psi\rangle = \alpha|\phi\rangle + \beta|\phi^\perp\rangle$, with the normalization condition $|\alpha|^2 + |\beta|^2 = 1$. This representation allows us to write the part of the output state whose control qubit is $|0\rangle$ as

$$\begin{aligned}
\frac{1}{2}|0\rangle(|\psi\rangle|\phi\rangle + |\phi\rangle|\psi\rangle) &= \frac{1}{2}|0\rangle(2\alpha|\phi\rangle|\phi\rangle + \beta|\phi^\perp\rangle|\phi\rangle + \beta|\phi\rangle|\phi^\perp\rangle) \\
&= |0\rangle(\alpha|\phi\rangle|\phi\rangle + \frac{\beta}{2}|\phi^\perp\rangle|\phi\rangle + \frac{\beta}{2}|\phi\rangle|\phi^\perp\rangle).
\end{aligned}$$

The acceptance probability, P_{acc} , given by measuring $|0\rangle$ is

$$\begin{aligned}
P_{acc}(|\psi\rangle, |\phi\rangle) &= |\alpha|^2 + \frac{1}{2}|\beta|^2 \\
&= |\alpha|^2 + |\beta|^2 - \frac{1}{2}|\beta|^2 \\
&= 1 - \frac{1}{2}(1 - |\alpha|^2) \\
&= \frac{1}{2} + \frac{1}{2}|\langle\phi|\psi\rangle|^2.
\end{aligned}$$

We formalize the swap test success probability in the following lemma for latter reference.

Lemma 4.5.1. *The swap test acceptance probability $P_{acc}(|\psi\rangle, |\phi\rangle)$, when applied to two unentangled pure states $|\psi\rangle$ and $|\phi\rangle$, is $P_{acc}(|\psi\rangle, |\phi\rangle) = \frac{1}{2} + \frac{1}{2}|\langle\phi|\psi\rangle|^2$.*

4.5.1 Swap Test for Mixed States

The swap test can also be applied to unentangled mixed states ρ and σ . To show the acceptance probability we adopt the ensembles view $\{\lambda_i, |i\rangle\}$ and $\{\lambda_j, |j\rangle\}$, arising from the spectral decompositions $\rho = \sum_i \lambda_i |i\rangle\langle i|$ and $\sigma = \sum_j \lambda_j |j\rangle\langle j|$.

$$\begin{aligned}
P_{acc}(\rho, \sigma) &= \sum_i \sum_j \lambda_i \lambda_j \left(\frac{1}{2} + \frac{1}{2} |\langle i|j \rangle|^2 \right) \\
&= \sum_i \sum_j \lambda_i \lambda_j \left(\frac{1}{2} + \frac{1}{2} \langle i|j \rangle \langle i|j \rangle \right) \\
&= \sum_i \sum_j \lambda_i \lambda_j \left(\frac{1}{2} + \frac{1}{2} \text{Tr}(|i\rangle\langle i| |j\rangle\langle j|) \right) \\
&= \frac{1}{2} + \frac{1}{2} \text{Tr} \left(\left(\sum_i \lambda_i |\psi\rangle\langle\psi| \right) \left(\sum_j \lambda_j |\phi\rangle\langle\phi| \right) \right) \\
&= \frac{1}{2} + \frac{1}{2} \text{Tr}(\rho\sigma)
\end{aligned}$$

Lemma 4.5.2. *The swap test acceptance probability $P_{acc}(\rho, \sigma)$, when applied to two unentangled mixed states $|\psi\rangle$ and $|\phi\rangle$, is $P_{acc}(|\psi\rangle, |\phi\rangle) = \frac{1}{2} + \frac{1}{2} \text{Tr}(\rho, \sigma)$.*

It is clear from above that even if the swap test is given two copies of the same mixed state ρ , the acceptance probability may be different than 1. A measure of ρ 's purity given by $\text{Tr}(\rho^2)$ may be much smaller than 1. In some cases, it might be interesting to detect whether a state is far from pure. This is another application of the swap test, as captured in the next lemma, where P_{rej} is the rejecting probability in the purity test.

Lemma 4.5.3. *Let ρ be a mixed state such that $\max_{|\xi\rangle} \langle \xi | \rho | \xi \rangle \leq 1 - \epsilon$. Then $\min_{\sigma} P_{rej}(\rho, \sigma) \geq \frac{\epsilon}{2}$.*

Proof. Using the bound in the hypothesis, the rejecting probability is

$$\begin{aligned}
\min_{\sigma} P_{rej}(\rho, \sigma) &= 1 - \max_{\sigma} P_{acc}(\rho, \sigma) \\
&= \frac{1}{2} - \max_{\sigma} \frac{1}{2} \text{Tr}(\rho, \sigma) \\
&= \frac{1}{2} - \max_{|\xi\rangle} \frac{1}{2} \langle \xi | \rho | \xi \rangle \\
&\geq \frac{\epsilon}{2}.
\end{aligned}$$

□

4.5.2 Swap Test for Entangled States

The analysis of the swap test assumes that the two input states are in a product form. However, it is worth investigating its behavior when an entangled state $|\xi\rangle^{AB}$ is given as input. For two unentangled states $|\psi\rangle$ and $|\phi\rangle$, after the application of the circuit corresponding to the swap test and just before measuring, the state in the accepting subspace is

$$|0\rangle(\alpha|\phi\rangle|\phi\rangle + \frac{\beta}{2}|\phi^\perp\rangle|\phi\rangle + \frac{\beta}{2}|\phi\rangle|\phi^\perp\rangle).$$

Let $|\xi\rangle^{AB} = \sum_k \lambda_k |k_A\rangle |k_B\rangle$ be in the Schmidt decomposition. Due to the linearity of the quantum operations so far, it is possible to analyze each state $\lambda_k |k_A\rangle |k_B\rangle$. Substituting $|\phi\rangle$ and $|\psi\rangle$ by $|k_A\rangle$ and $|k_B\rangle$, respectively, in the previous state, we have

$$\lambda_k |0\rangle(\alpha|k_A\rangle|k_A\rangle + \frac{\beta}{2}|k_A^\perp\rangle|k_B\rangle + \frac{\beta}{2}|k_B\rangle|k_A^\perp\rangle).$$

The swap test accepting probability is

$$P_{acc}(|\xi\rangle^{AB}) = \sum_k \lambda_k^2 \left(\frac{1}{2} + \frac{1}{2} |\langle k_A | k_B \rangle|^2 \right) = \sum_k \lambda_k^2 P_{acc}(|k_A\rangle, |k_B\rangle).$$

Lemma 4.5.4. *The swap test acceptance probability $P_{acc}(|\psi\rangle)$ when applied to an entangled state $|\psi\rangle = \sum_k \lambda_k |k_A\rangle |k_B\rangle$ given by its Schmidt decomposition is $P_{acc}(|\psi\rangle) = \sum_k \lambda_k^2 P_{acc}(|k_A\rangle, |k_B\rangle)$.*

The maximally entangled state is an example of state that causes this test to accept with probability 1. Nevertheless, it is interesting to observe that this test does enforce a certain structure on the entangled state $|\xi\rangle^{AB}$. In the Schmidt decomposition there is no need for $|k_A\rangle$ and $|k_B\rangle$ to have a large overlap. If this is not the case for values of k corresponding to big λ_k , then the swap test acceptance probability decreases.

4.6 Quantum Fourier Transform

A quantum operation on n qubits can be represented as a $2^n \times 2^n$ unitary matrix. The problem is that there is no guarantee that quantum operations admit an efficient implementation using only gates that act on at most two qubits from a universal set. The Quantum Fourier Transform (QFT) was no exception, until Shor gave an efficient implementation for it [91] [74]. Contrary to the classical Discrete Fourier Transform (DFT) which receives the explicit description of a signal and outputs the amplitude of all frequencies, the QFT works implicitly with the amplitudes of a quantum state. The result is another quantum state whose amplitude frequencies are encoded in the quantum state itself. Assessing the frequencies can be done indirectly by measuring the state. The frequencies with higher amplitudes will have a higher probability of being an outcome. For this reason, the QFT is also known as Fourier sampling. In this regard, the QFT and DFT accomplish different tasks. The n -qubit QFT unitary F_{2^n} can be represented by the following matrix

$$F_{2^n} = \frac{1}{\sqrt{2^n}} \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{2^n-1} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(2^n-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{(2^n-1)} & \omega^{2(2^n-1)} & \cdots & \omega^{(2^n-1)(2^n-1)} \end{pmatrix},$$

where $\omega = e^{\frac{2\pi i}{2^n}}$. To understand why the QFT admits an efficient implementation, we analyze its action on a basis state $|y\rangle$ and show how the resulting state can be wisely rewritten in product form. The state $F_{2^n}|y\rangle$ is simply the y^{th} column of F_{2^n} . Representing the frequencies in binary form $x = x_1x_2 \dots x_n$, this state is

$$F_{2^n}|y\rangle = \sum_{x_1x_2\dots x_n \in \{0,1\}^n} e^{\frac{2\pi i y x_1x_2\dots x_n}{2^n}} |x_1x_2\dots x_n\rangle = \sum_{x_1x_2\dots x_n \in \{0,1\}^n} e^{2\pi i y 0.x_1x_2\dots x_n} |x_1x_2\dots x_n\rangle.$$

Shor made the important observation that this state can be rewritten as

$$\frac{(|0\rangle + e^{2\pi i y 0.1}|1\rangle)}{\sqrt{2}} \otimes \frac{(|0\rangle + e^{2\pi i y 0.01}|1\rangle)}{\sqrt{2}} \otimes \cdots \otimes \frac{(|0\rangle + e^{2\pi i y 0.00\dots 1}|1\rangle)}{\sqrt{2}}.$$

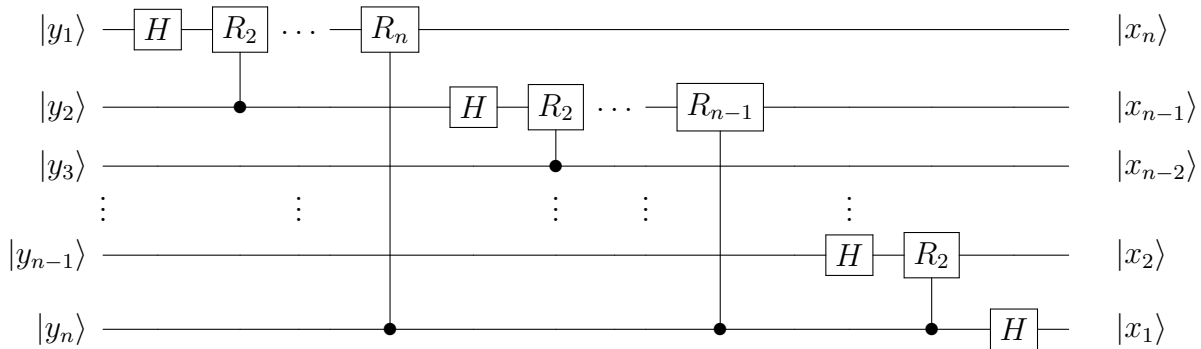
Exploring the binary notation of the input state $y = y_1y_2\dots y_n$ and the equivalence $e^{2\pi i a.b} = e^{2\pi i 0.b}$ where $a.b$ is a floating binary number, we have

$$\frac{(|0\rangle + e^{2\pi i 0.y_n}|1\rangle)}{\sqrt{2}} \otimes \frac{(|0\rangle + e^{2\pi i 0.y_{n-1}y_n}|1\rangle)}{\sqrt{2}} \otimes \cdots \otimes \frac{(|0\rangle + e^{2\pi i 0.y_1y_2\dots y_n}|1\rangle)}{\sqrt{2}}.$$

The efficient QFT implementation uses a controlled version of the phase shift gate R_k :

$$R_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{2^k}} \end{pmatrix}.$$

The efficient QFT implementation is shown next. It uses $O(n^2)$ R_k gates and $O(n)$ Hadamard gates.



This analysis done for a basis state extends to an arbitrary state due to linearity. The QFT is a crucial component in the famous Shor's factoring algorithm. It also plays an important role in complexity theory to ensure that a state received from a prover has a certain structure.

4.7 Simulating k -Merlins with Two

Harrow and Montanaro showed that a test called the product test P_{test} can check whether two unentangled states are close to a product on k subsystems. With this test, it is possible to show the collapse $\text{QMA}(k) = \text{QMA}(2)$. First, we need a simple lemma.

Lemma 4.7.1 (From [51]). *Let $|\psi\rangle, |\phi\rangle$ be pure states such that $|\langle\psi|\phi\rangle|^2 = 1 - \epsilon$, and let M satisfy $0 \leq M \leq I$. Then $|\langle\psi|M|\psi\rangle - \langle\phi|M|\phi\rangle| \leq \sqrt{\epsilon}$.*

Proof. The trace distance for pure states $|\psi\rangle$ and $|\phi\rangle$ is

$$\frac{1}{2} \|\psi\rangle\langle\psi| - |\phi\rangle\langle\phi|\|_1 = \sqrt{1 - |\langle\psi|\phi\rangle|^2} = \sqrt{\epsilon}.$$

The trace distance can also be formulated as the optimization problem

$$\max_{P: 0 \leq P \leq I} \text{Tr}(P(|\psi\rangle\langle\psi| - |\phi\rangle\langle\phi|)) = \frac{1}{2} \|\psi\rangle\langle\psi| - |\phi\rangle\langle\phi|\|_1.$$

As M satisfies $0 \leq M \leq I$, it is clear that $\text{Tr}(M(|\psi\rangle\langle\psi| - |\phi\rangle\langle\phi|))$ gives a lower bound on this distance and the lemma follows. \square

The collapse lemma is presented next.

Lemma 4.7.2 (From [51]). *For any m, k , $0 \leq s < c \leq 1$,*

$$\text{QMA}(k, c, s)_m \subseteq \text{QMA}(2, c', s')_{km},$$

where $c' = \frac{1+c}{2}$ and $s' = 1 - \frac{(1-s)^2}{100}$

To show the previous lemma Harrow and Montanaro used a protocol that with equal probability performs the product test or performs the original $\text{QMA}(k)$ protocol.

- 1 **Desired Input:** Receive the same state $|\psi\rangle = |\psi_1\rangle \otimes \dots \otimes |\psi_k\rangle$ from both provers.
- 2 With equal probability perform one of the test:
- 3 1) (Close to Product)
- 4 Perform the product test accepting iff it accepts;
- 5 2) (Original Protocol)
- 6 Choose one of the proofs uniformly at random;
- 7 Perform the original $\text{QMA}(k)$ protocol with the chosen proof accepting iff it accepts;

Algorithm 3: Simulating $\text{QMA}(k)$ in $\text{QMA}(2)$.

Proof. The completeness is straightforward. If the two provers send the same state $|\psi\rangle = |\psi_1\rangle \otimes \cdots \otimes |\psi_k\rangle$ that causes Arthur to accept with probability at least c , the product test will accept with probability 1. Then, the total accepting probability at least $\frac{1+c}{2}$ as the product test and the original protocol are performed with equal probability.

For the soundness, we assume that the two provers sent $|\phi_1\rangle$ and $|\phi_2\rangle$ whose maximal overlap with a product state is $1 - \epsilon_1$ and $1 - \epsilon_2$, respectively. Also, let the product test accept with probabilities $P_{test}(\phi_1) = 1 - \delta_1$ and $P_{test}(\phi_2) = 1 - \delta_2$. Set $\epsilon = \frac{\epsilon_1 + \epsilon_2}{2}$ and $\delta = \frac{\delta_1 + \delta_2}{2}$. The product test when performed with $|\phi_1\rangle$ and $|\phi_2\rangle$ accepts with probability

$$\frac{1}{2^k} \sum_{S \subseteq [k]} \text{Tr}(\phi_1 \phi_2)_S \leq \frac{1}{2^k} \sum_{S \subseteq [k]} \sqrt{\text{Tr}(\phi_1)_S^2} \sqrt{\text{Tr}(\phi_2)_S^2},$$

where $\text{Tr}(\rho)_S = \text{Tr}(\text{Tr}_{S^c}(\rho))$. Using the arithmetic mean geometric mean inequality ($\frac{x_1 + \cdots + x_n}{n} \geq \sqrt[n]{x_1 \cdots x_n}$) gives

$$\begin{aligned} \frac{1}{2^k} \sum_{S \subseteq [k]} \text{Tr}(\phi_1 \phi_2)_S &\leq \frac{1}{2^k} \sum_{S \subseteq [k]} \frac{\text{Tr}(\phi_1)_S + \text{Tr}(\phi_2)_S}{2} \\ &= \frac{1}{2} (P_{test}(\phi_1) + P_{test}(\phi_2)) \\ &= 1 - \frac{1}{2} (\delta_1 + \delta_2) = 1 - \delta. \end{aligned}$$

□

Chapter 5

Complexity Classes

A language L is simply a set of finite words (strings) of the form $x \in \Sigma^*$ that were arbitrarily grouped together. The set Σ contains the symbols used to create words, for this reason, it is denoted the alphabet. In complexity theory, we typically have $\Sigma = \{0, 1\}$ as there is no asymptotic advantage of working with larger alphabets when they contain a constant number of elements. A language can be seen as partitioning the set of strings into two sets L_{yes} and L_{no} where the first one represents strings in the language whereas the second one is its complement. Instances of problems can be represented in binary using some appropriate encoding scheme. For a decision problem (whose output is just “yes” or “no”) and an encoding, it is possible to associate a language consisting of encoded instances of accepting inputs. It is usually assumed that instances are encoded with an efficient encoding which prevents its binary representation from being unnecessarily large. With this notion in mind, computational resources required to decide if a given string belongs to a language are often measured in terms of the input size. It is natural to think that different problems will require different amounts of resources. In this context, a complexity class definition presupposes a precise computational model and imposes resources constraints on it. This combination might be sufficient to decide a variety of languages which are said to belong to the complexity class.

Since the inception of complexity theory, a myriad of complexity classes were proposed as a way to understand fundamental computational resources such as time and space. The concept of proof verification became paramount in this theory. Randomness found its way in several settings from proof verification to interactions with adversarial provers. Currently, there is a reasonable understanding about the inclusion relationships among complexity classes or, equivalently, how some resources can be traded by others. Apart from a complex network of inclusions, there are few proofs that actually establish unconditional separations between classes. Part of the reason for this is that even resource constrained computational models can be so expressive that it is hard to rule out

languages they can not decide. As an alternative, many conjectures regarding unproven separations were proposed. Some of them are supported by strong evidence gathered by theorists. The claim $P \neq NP$ is treated as almost an unquestionable truth in the complexity community while conjectures such as the Unique Games Conjecture (UGC) [93] and the Quantum PCP Conjecture divide opinions.

Classical computational complexity studies questions such as the limits of efficient computation in the abstract setting of Turing machines. The Church-Turning Thesis (CTT) stipulates that every effective computation can be realized by a Turing Machine [92]. In turn, the Extended Church-Turning Thesis (ECTT) goes beyond this by stating that any effective computation can be simulated with at most a polynomial slow down in a Turing Machine when compared to other physical realizable models. The existence of Shor's factoring algorithm which places factoring in the realm of quantum polynomial time is a threat to the ECTT. Clearly, factoring is in NP. Despite being conjectured intractable, it is not believed to be NP-complete. In some sense, it is believed to be an intermediate problem between P and the hardest problems in NP.

In the quest for the limits of efficient computation, quantum computing provides a more concrete picture of the physical world as it relies on the postulates of quantum mechanics. This theory is one of the most well tested and accurate physics theory. It captures our foremost understanding of nature at small scales.

In this chapter, we give an overview of important quantum and classical complexity classes. We build on Watrous work providing an updated version of his complexity class diagram [96]. We present some computational resources and how they fit in this complexity diagram. The main goal is to give a precise definition of $QMA(k)$ and provide information for close complexity classes. For further information, the complexity zoo project collects an extensive list of complexity classes [1].

5.1 Efficient Computation

The notion of efficient computation is usually studied in the asymptotic regime when running time is measured as function of the input length. Algorithms whose running time can be bounded by a polynomial are considered efficient regardless of constants and polynomial degree. On the other side of the spectrum, the standard notion of intractability for a problem requires the minimum running time to be exponential. Note that with only these two notions problems in between are left unclassified.

The quest for polynomial time algorithms was pioneered by Jack Edmonds. Since then, the notion of an efficient computation was extended to a variety of computational models. One of the simplest models of general purpose computation is the deterministic Turing Machine. Problems solved in polynomial time in this model give rise to the famous

class P which constitutes our classical notion of efficient computation. This class can be simply defined as follows.

Definition 5.1.1 (P). *A language L is in P if there is a deterministic polynomial time Turing machine V satisfying*

- *if $x \in L$, then V accepts x .*
- *if $x \notin L$, then V rejects x .*

One simple way to make a Turing Machine more flexible is to extend its transition function to become a relation in which each transition has an associated probability, described by polynomially many bits. This machine is known as a Probabilistic Turing Machine. It turns out that this new model is equivalent to a deterministic one with access to a polynomial size random bit string. Currently, it is not known if a deterministic machine can simulate a probabilistic one without access to a random string. In other words, it is not known if derandomization is possible in this model. It might be the case that randomness makes the machine more powerful, turning it into an important resource. The notion of efficient probabilistic computation is captured by the Bounded-error Probabilistic Polynomial time (BPP) class.

Definition 5.1.2 (BPP). *A language L is in BPP if there is a probabilistic polynomial time Turing machine V satisfying*

- **Completeness:** *if $x \in L$, $\Pr[V \text{ accepts}] \geq \frac{2}{3}$.*
- **Soundness:** *if $x \notin L$, $\Pr[V \text{ accepts}] \leq \frac{1}{3}$.*

The Polynomial Hierarchy (PH) is a family of complexity classes that can be inductively defined in terms of oracles as follows.

- $\Delta_0 P = \Sigma_0 P = \Pi_0 P = P$.
- For $i > 0$, we have:
 - $\Delta_i P = P^{\Sigma_{i-1} P}$;
 - $\Sigma_i P = NP^{\Sigma_{i-1} P}$; and
 - $\Pi_i P = \text{coNP}^{\Sigma_{i-1} P}$.

It is conjectured that this hierarchy does not collapse to any finite level [85].

The class BPP is contained in the first levels of the Polynomial hierarchy, more precisely, in $\Sigma_2 P \cap \Pi_2 P$ [10]. This result puts an upper bound on the extra power provided by randomness. Moreover, if for each input size a deterministic Turing Machine receives

a chosen fixed string from a trusted source, then this new model contains BPP. In complexity theory, this extra string receives the name of an advice. The result just mentioned can be stated as $\text{BPP} \subseteq \text{P}_{/poly}$ [63].

Before introducing the notion of efficient quantum computation, we need to establish how quantum computations can be realized. Abstractly, a quantum verification procedure can be modeled as an unitary operation circuit V_x that acts on an ancilla state $|0\rangle^{\otimes m}$ followed by a projective measurement $\{\Pi_{acc} = |0\rangle\langle 0|, \Pi_{rej} = \mathbb{I} - \Pi_{acc}\}$ to decide whether to accept or reject. The issue with this abstraction is that V_x is arbitrary, but in practice we may only have a fixed number of local unitaries at our disposal. Each local unitary will have a unity time cost. Furthermore, to actually build the circuit from these local operations we need its classical description. For an input x , this description must be generated efficiently by a classical Turing Machine. A family of circuits $\{V_x\}_x$ whose description can be efficiently generated by this kind of machine is denoted polynomial time uniformly generated. Finally, polynomial time quantum computations can be realized by this kind of family.

With the appropriate differences, the class Bounded-error Quantum Polynomial time (BQP) is defined in an analogous way to the BPP class.

Definition 5.1.3 (BQP). *A language L is in BQP if there is a family of polynomial time uniformly generated quantum verifiers $\{V_x\}_x$ satisfying*

- **Completeness:** *if $x \in L$, $\langle 0|^{\otimes m} V_x^\dagger \Pi_{acc} V_x |0\rangle^{\otimes m} \geq \frac{2}{3}$.*
- **Soundness:** *if $x \notin L$, $\langle 0|^{\otimes m} V_x^\dagger \Pi_{acc} V_x |0\rangle^{\otimes m} \leq \frac{1}{3}$.*

Where $m = p(|x|)$ for some polynomial p .

5.2 Proof Verification

Some problems appear to be intractable in a deterministic Turing Machine. However, when given a proof from a computationally unbounded adversarial entity, which we call a prover, these problems admit an efficient verification. For this reason, proof verification plays a central role in complexity theory.

5.2.1 Classical Proof Verification

The complexity class NP captures the languages that can be efficiently *i.e.*, in polynomial time, decided by a classical deterministic verifier. This class can be equivalently defined in terms of non-deterministic polynomial time Turing Machines [41].

Definition 5.2.1 (NP). *A language L is in NP if there exists a deterministic polynomial time verifier V and a polynomial p such that*

- *if $x \in L$, then there exists a witness $y \in \{0, 1\}^{p(|x|)}$ such that $V(x, y)$ accepts;*
- *if $x \notin L$, then for all $y \in \{0, 1\}^{p(|x|)}$ $V(x, y)$ rejects.*

In the above statement, the string y is referred to interchangeably as the proof, the witness, or the certificate. It attests the membership of x in the language L , and can be thought of as being given to the verifier by a computationally unbounded prover.

The boolean formula satisfiability problem, known as SAT, is of central importance to NP. It is defined as follows.

Theorem 5.2.2 (SAT). *The SAT language is composed of instances φ in propositional conjunctive normal form (CNF), and that have at least one satisfying assignment.*

Clearly, SAT is in NP as a satisfying assignment has polynomial size and can be used as a membership certificate. Moreover, all languages in NP can be reduced to SAT in polynomial time making it a NP-hard problem. Combining these two results, we have the Cook-Levin Theorem which is arguably the most important theorem in computational complexity.

Theorem 5.2.3 (Cook-Levin [41]). *SAT is NP-complete.*

There are many other NP-complete problems. The 3SAT is a restricted version of SAT in which each clause has three literals. Even though it is more constrained, it is also NP-complete. More broadly, we can think about Constraint Satisfaction Problems (CSP) in which the constraints act on q variables that assume values in an arbitrary alphabet Σ . Each constraint can be viewed as a function $f : \Sigma^q \rightarrow \{0, 1\}$ indicating which tuple of values are satisfiable. This problem is denoted $q\text{-CSP}_\Sigma$. Graph three Coloring (3COL) is a 2-CSP on a ternary alphabet. This language is also NP-complete, as several other variations of CSPs are as well. One important exception is 2SAT which is in P.

The classes NP has a scaled-up version known as NEXP in which the certificates may have exponential size.

Definition 5.2.4 (NEXP). *A language L is in NEXP if there exists an exponential time deterministic verifier V and a polynomial p such that*

- *if $x \in L$, then there exists a witness $y \in \{0, 1\}^{2^{p(|x|)}}$ such that $V(x, y)$ accepts;*
- *if $x \notin L$, then for all $y \in \{0, 1\}^{2^{p(|x|)}}$ $V(x, y)$ rejects.*

5.2.2 Quantum Proof Verification

The description of NP by way of proof verification was also extended to the quantum model, leading to the class QMA. This class extends NP by allowing the witness and the verification procedure to be both quantum. In fact, since quantum computation is inherently probabilistic its closest classical analogue is MA which is an extension of NP with a probabilistic verifier.

Definition 5.2.5 (QMA). *A language L is in $\text{QMA}(c(n), s(n))$ if there are polynomial time computable functions $c, s : \mathbb{N} \rightarrow [0, 1]$, such that for every $x \in \{0, 1\}^n$ there is a polynomial time quantum verifier V_x satisfying*

- **Completeness:** *if $x \in L$, there is a witness $|\psi\rangle$ with $\Pr[V_x|\psi\rangle \text{ accepts}] \geq c(n)$;*
- **Soundness:** *if $x \notin L$, then for all $|\psi\rangle$ $\Pr[V_x|\psi\rangle \text{ accepts}] \leq s(n)$.*

Moreover, the witness has $p(n)$ qubits for some polynomial p and the completeness soundness gap, $c(n) - s(n)$, is at least $\frac{1}{g(n)}$ for some polynomial $g(n)$.

In the previous definition, given that the fraction $\frac{1}{g(n)}$ is an inverse polynomial, it is possible to amplify a QMA protocol to achieve exponentially small completeness and soundness errors without changing the witness size [69]. Even though completeness and soundness can be arbitrary, it is common to define QMA as $\text{QMA}(\frac{2}{3}, \frac{1}{3})$.

A promise problem is similar to a language, but instead of inducing a bipartition of all strings over a given alphabet it just specifies two disjoint sets $(L_{\text{yes}}, L_{\text{no}})$ whose union is not required to be the whole set strings. In the quantum community, promise problems are extensively used. The definitions provided here can be easily adapted to promise problems instead of languages. What is interesting about this formalism is that it provides a way of establishing instance properties. The verification behaviour on strings not in $L_{\text{yes}} \cup L_{\text{no}}$ is not important. Here, we work with both formalisms.

The quantum analogue of k -CSP is the k -Local Hamiltonian (k -LH) problem. It is defined as follows.

Definition 5.2.6 (k -LH). *Given a list of m k -local operators $H_i \in \text{Herm}(C^{d^k})$ acting on n particles each of dimension $d \in O(1)$ and polynomial time computable functions $a, b : \mathbb{N} \rightarrow \mathbb{N}$, the promise problem k -Local Hamiltonian consists in distinguishing the cases*

- $H \in L_{\text{yes}}$, then there is a state $|\psi\rangle$ such that $\langle\psi|H|\psi\rangle \leq a$;
- $H \in L_{\text{no}}$, then for every state $|\psi\rangle$ it holds that $\langle\psi|H|\psi\rangle \geq b$;

where $H = \sum_{i=1}^m H_i$, and $b - a \geq \frac{1}{p(n)}$ for some polynomial p .

The k -LH plays a similar role in quantum proof verification as the classical CSPs do in classical proof verification. There is a quantum analogue to the celebrated Cook-Levin Theorem.

Theorem 5.2.7 (Kitaev Quantum Cook-Levin [63]). *k -LH is QMA-complete.*

In this work, we are particularly interested in the extension of QMA that uses multiple unentangled provers, denoted $\text{QMA}(k)$. Observe that the study of multi-prover MA is non-trivial only in the quantum model where the unentanglement promise plays a crucial role [4]. This distinctive quantum class is formally defined next.

Definition 5.2.8 ($\text{QMA}(k)$). *A language L is in $\text{QMA}(k, c(n), s(n))_{l(n)}$ if there are polynomial time computable functions $c, s : \mathbb{N} \rightarrow [0, 1]$, $l : \mathbb{N} \rightarrow \mathbb{N}$, such that for every $x \in \{0, 1\}^n$ there is a polynomial time quantum verifier V_x satisfying*

- **Completeness:** *if $x \in L$, then there is a witness $|\psi_1\rangle \otimes \cdots \otimes |\psi_k\rangle$ with $\Pr[V_x|\psi_1\rangle \otimes \cdots \otimes |\psi_k\rangle \text{ accepts}] \geq c(n)$.*
- **Soundness:** *if $x \notin L$, then for all $|\psi_1\rangle \otimes \cdots \otimes |\psi_k\rangle$ $\Pr[V_x|\psi_1\rangle \otimes \cdots \otimes |\psi_k\rangle \text{ accepts}] \leq s(n)$.*

Moreover, for each $i \in [k]$ the witness size is bounded by $O(l(n))$ qubits and the completeness soundness gap, $c(n) - s(n)$, is at least $\frac{1}{g(n)}$, for some polynomial $g(n)$.

5.3 Statistical Zero Knowledge

The Graph Non-Isomorphism (GNI) problem asks whether two graphs (G_1, G_2) are non-isomorphic. It seems hard to conceive a classical polynomial size proof certifying that there is no possible isomorphism between them. However, a simple protocol in Statistical Zero Knowledge (SZK) is enough to attest with high probability whether they are isomorphic. The class SZK provides a minimalist interaction with a prover in which the verifier may learn nearly nothing other than it could have computed itself.

The Statistical Distance (SD) is a complete problem for the class SZK. It is defined next where C_0 and C_1 are classical circuits.

Definition 5.3.1 (From [87]). *Statistical Distance (SD) is the promise problem $L = (\text{SD}_{\text{yes}}, \text{SD}_{\text{no}})$ such that*

$$\text{SD}_{\text{yes}} = \{(C_0, C_1) : \|C_0, C_1\|_1 > 2/3\}$$

$$\mathbf{SD}_{no} = \{(C_0, C_1) : \|C_0, C_1\|_1 < 1/3\}$$

where $\|C_0, C_1\|_1$ is the statistical difference of their outputs when given a uniform distribution on the inputs.

Note that using to an amplification lemma in [87], it is possible to assume w.l.o.g. that $\|C_0, C_1\|_1 > 1 - 2^{-poly}$ for a yes instance, and that $\|C_0, C_1\|_1 < 2^{-poly}$ for a no instance.

The GNI and **SD** problems are interesting candidate problems to place in QMA(2). A proof of this fact might indicate that QMA(2) is larger than QMA.

5.4 Interaction

Having a proof of a statement may be much easier than proving it yourself, as captured by the conjecture $P \neq NP$. Intuitively, having the opportunity to talk with a skillful prover inquiring why a certain statement might be true is apparently more powerful than being given a proof alone. In complexity terms, the set of languages admitting such proof system is naturally denoted Interactive Proof System (IP), as interaction is the fundamental resource. It is usually parameterized as $IP[k]$ where k is the total number of messages exchanged between the prover and verifier. To be more expressive than NP, the underlying computational model of IP needs another resource. Imagine that the verifier is a deterministic procedure. In this case, the prover can anticipate all of the verifier's questions, and it could alternatively send all answers at once making IP equivalent to NP. Consequently, randomness is also a crucial resource, making future questions to the prover hard to predict.

As it is typical in complexity theory, the prover is an computationally unbounded adversarial entity. It would try to maximize the acceptance probability in all cases, even though the input is not in the language. With interactions and randomness, the verifier can make the prover commit to some answers without letting the prover know precisely what questions might come next. Intuitively, this system seems more powerful to detect a cheating prover. The general view of an $IP[k]$ protocol for even k is shown next.

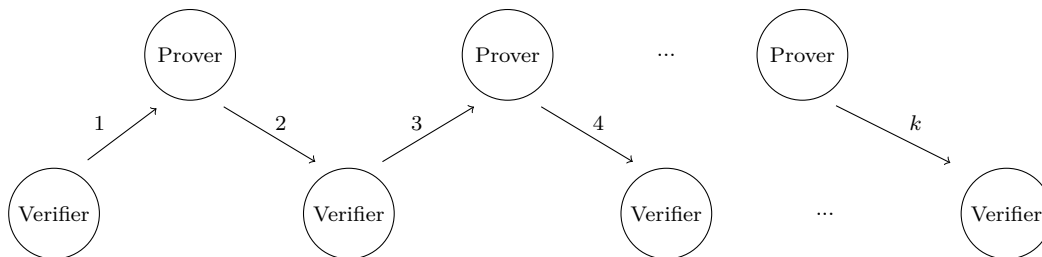


Figure 5.1: IP with an even number of messages.

Given a verifier, it is easy to create a brute-force simulation that tests all possibilities of prover's strategies over random strings for the verifier in polynomial space (PSPACE). The converse statement may not be evident, but it is also true, i.e. $\text{PSPACE} \subseteq \text{IP}$. This result was a major breakthrough in complexity theory.

Theorem 5.4.1 (From [89]).

$$\text{PSPACE} \subseteq \text{IP}$$

In the definition of IP, the randomness used by the verifier is private and the messages are arbitrary strings. It is possible to restrict this model by making the verifier's messages the random bit strings it is going to use. This notion of public coins leads to the definition of the Arthur-Merlin class (AM).

Definition 5.4.2 (AM). *The class $\text{AM}[k]$ is a restricted version of $\text{IP}[k]$ in which verifier's messages consist only of random bits and they are the only random bits the verifier is allowed to use.*

Goldwasser and Sipser showed a surprising result that private coins IP do not provide an asymptotic advantage in the number of rounds over public coins. In fact, any private coin IP system with k messages can be replaced by an AM system with two extra messages.

Theorem 5.4.3 (From [44]). *For every $k : \mathbb{N} \rightarrow \mathbb{N}$ with $k(n)$ computable in polynomial time in n ,*

$$\text{IP}[k] = \text{AM}[k + 2].$$

Quantum interactive proof systems (QIP) are defined in a similar way to IP, except that the verifier and prover are quantum machines that may exchange quantum messages. Surprisingly, the class QIP is equal to IP.

Theorem 5.4.4 (Adapted from [58]).

$$\text{QIP} = \text{IP}.$$

However, there is one important distinction between these systems. Any quantum interactive proof system with polynomially many messages can be transformed into another with just three messages. This result is not possible classically, unless the polynomial hierarchy collapses. This is another strong indication that quantum information can be used to perform tasks which are impossible to perform classically.

Theorem 5.4.5 (Kitaev and Watrous [62]).

$$\text{QIP} \subseteq \text{QIP}(3).$$

In terms of circuits, we can think of the underlying computational model of QIP as a multi-partite register in which the prover and the verifier unitaries act and, after all communication is done, the verifier decides to accept or reject by measuring one of its qubits. This register is a combination of three smaller ones: the private space of the verifier V , a common message register M through which the messages are exchanged, and the private space of the prover P . The number of qubits in V and M is bounded by a polynomial whereas, in principle P , can contain arbitrarily many qubits. We consider three messages protocols since they are sufficient to capture the whole QIP class power. In this case, the verifier has two unitaries, V_1 and V_2 , acting exclusively on M and V . These verifier's unitaries must be implementable with polynomially many local unitaries from a fixed constant size universal set. The prover's goal is to devise unitaries P_1 and P_2 acting exclusively on M and P to maximize the verifier's acceptance probability. The final circuit is represented next. Without loss of generality, we assume that the registers begin in the computational basis $|0\rangle$.

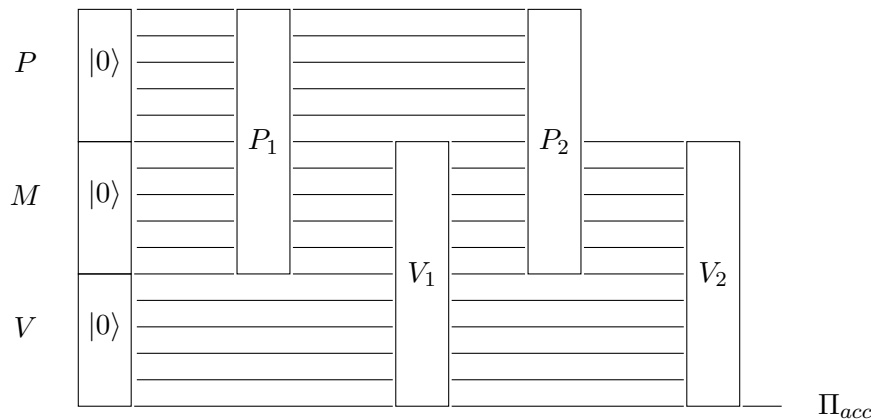


Figure 5.2: Circuit view of a QIP(3) protocol.

Currently, the precise power of QIP(2) is a central question in quantum interactive proof systems. We do not know whether it is strictly smaller than QIP(3) or not.

5.5 Refereed Games

The prover in the definition of IP will always try to convince the verifier that a certain statement is true. A natural extension to this model occurs with the addition of another prover whose goal is to convince the verifier of the opposite. Having both a “yes” and “no” prover, the verifier has the certainty that at least one of them will be truthful. This model gives rise to the computational class Refereed Games (RG) in which the verifier is denoted referee and the provers are denoted “Yes Player” and “No Player”. The game proceeds

in rounds, where each one is composed by a question sent to each player in parallel as well as their responses. Similarly to IP, it is also parameterized but now messages sent in parallel counts as one. An example of a RG round is illustrated next.

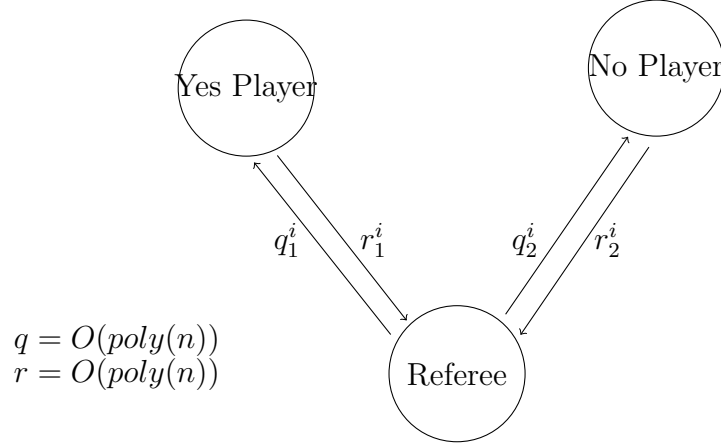


Figure 5.3: The i^{th} round of a Refereed Game.

This new resource of a “No Player” seems non-trivial. In fact, RG with polynomially many rounds is equal to EXP which is conjectured to be a bigger class than PSPACE.

Theorem 5.5.1 (From [34]).

$$\text{RG} = \text{EXP}.$$

Refereed games extend in the natural way to the quantum model, being known as Quantum Refereed Games (QRG). In this new class, the players and verifiers are now quantum machines exchanging quantum messages. It is possible to assume that the players do not share entanglement because of the adversarial nature of the players. For this reason, QRG could be a good candidate for upper bounding QMA(2). However, it is important to be careful because the reduction can not simulate a generic QMA(2) protocol within an inverse exponential additive error, in the acceptance probability, in exponential time unless EXP = NEXP.

A pattern that starts to become recurrent in classes bigger than QMA is that using quantum information does not increase the computational power for some of them. This case is no exception, the classes RG and QRG are the same.

Theorem 5.5.2 (From [49]).

$$\text{QRG} = \text{EXP}.$$

5.6 Multi-prover

A resource that seems more powerful than having two provers with antagonistic roles, as in RG, is to have multiple collaborative provers that are not allowed to communicate with each other. In a metaphorical perspective, this model gives the verifier the opportunity to interrogate the provers independently and confront their answers. This model is so powerful that it suffices to interrogate just two provers in parallel with polynomial size questions which are replied to with constant size answers, in order to verify any language in NEXP. This Multi-Prover Interactive Systems Proof model gives rise to the class $\text{MIP}(k)$, where k is the number of provers. A high-level view of the protocol just described is illustrated next.

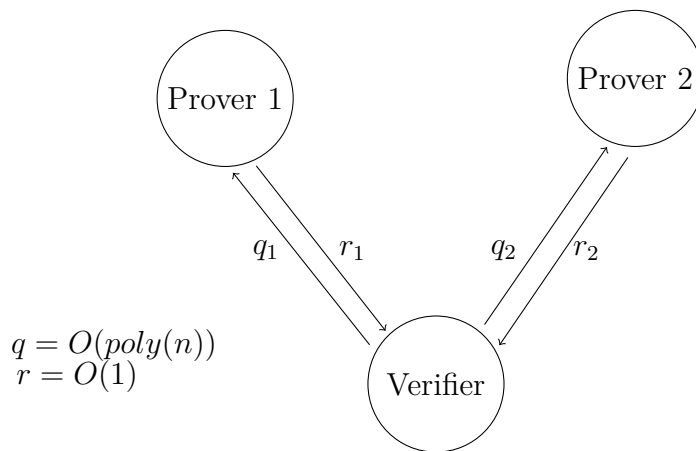


Figure 5.4: One round is enough to show $\text{NEXP} \subseteq \text{MIP}(2)$.

It is easy to see that MIP is upper bounded by NEXP, as each player strategy can be viewed as an exponential size proof since the questions have polynomial size and the number of rounds is bounded by a polynomial. Therefore, MIP reaches its full power with a very simple protocol.

The closest quantum generalization of MIP is QMIP_{ne} . The *ne* subscript stands for not entangled. In both cases, the provers can not communicate neither share entangled states. The exchanged messages and computations are quantum in QMIP_{ne} . Clearly, we have $\text{MIP} \subseteq \text{QMIP}_{ne}$. Moreover, the class QMIP_{ne} is also upper bounded by NEXP, and so the classical and quantum variations are actually the same class.

It is possible to generalize the classical MIP to provers that share arbitrary entangled states while the messages remain classical. This new version is known as MIP^* and, contrary to our intuition, entanglement among the players does not make it smaller than MIP. It does not improve the player's ability to collude. In fact, Vidick and Ito showed that $\text{NEXP} \subseteq \text{MIP}^*$ [57]. Allowing the messages to be quantum, we have a variation

of MIP^* denoted QMIP . Once again, the quantum and classical classes are the same *i.e.* $\text{MIP}^* = \text{QMIP}$ [84]. Unlike 2-player **XOR** games whose entangled state admit a characterization (Tsirelson), the entangled states used in MIP^* or QMIP can live in a Hilbert Space of arbitrarily large dimension. The lack of bounds for this dimension makes it hard to upper bound these classes. It is a viable possibility that shared entanglement can make MIP^* more powerful than MIP .

5.7 Complexity Class Diagram

Extending Watrous complexity class diagram [96] with recent developments, we have the Diagram 5.5.

There are important observations that can be drawn from this diagram. Note that the precise power of $\text{QMA}(2)$ is largely unknown. It is only known its trivial bounds: QMA and NEXP . We make again the observation that some “powerful” classical classes are equal to their quantum analogues. In some sense, their associated computational model becomes so powerful that quantum information is not able to increase the class any further. Nevertheless, as in the case of QIP , surprising differences may occur as three quantum messages are enough to capture the power of this entire model. Even though large scale quantum computation may turn out to be impossible, the study of theoretical quantum computing is of independent interest, besides giving rise to beautiful mathematics. Proving that $\text{BQP} \neq \text{BPP}$ would imply the unknown important classical result $\text{P} \neq \text{PSPACE}$.

5.8 Hierarchies

In computational complexity, we have few unconditional separations between classes.

5.8.1 Time Hierarchy

$$T_f = \{ \langle M, x \rangle \mid M \text{ accepts } x \text{ in at most } f(n) \text{ steps for } n = | \langle M, x \rangle | \}$$

Suppose there is a machine M' that decides the language T_f in at most $f(\lfloor \frac{n}{2} \rfloor - 1)$ time steps where n is the input size. We create a diagonalization machine D_f using M' as follows.

We run D_f over its own description. Let $n = |D_f|$ be the input size in this case. If M' accepts the input $\langle D_f, D_f \rangle$, then D_f accepts its own code in time at most $f(2n)$. By hypothesis, M' ran in time at most $f(\lfloor \frac{2n}{2} \rfloor - 1)$, thus D_f can be simulated in time at most $f(n)$. This is a contradiction since $D_f(D_f)$ rejected its own description in time at

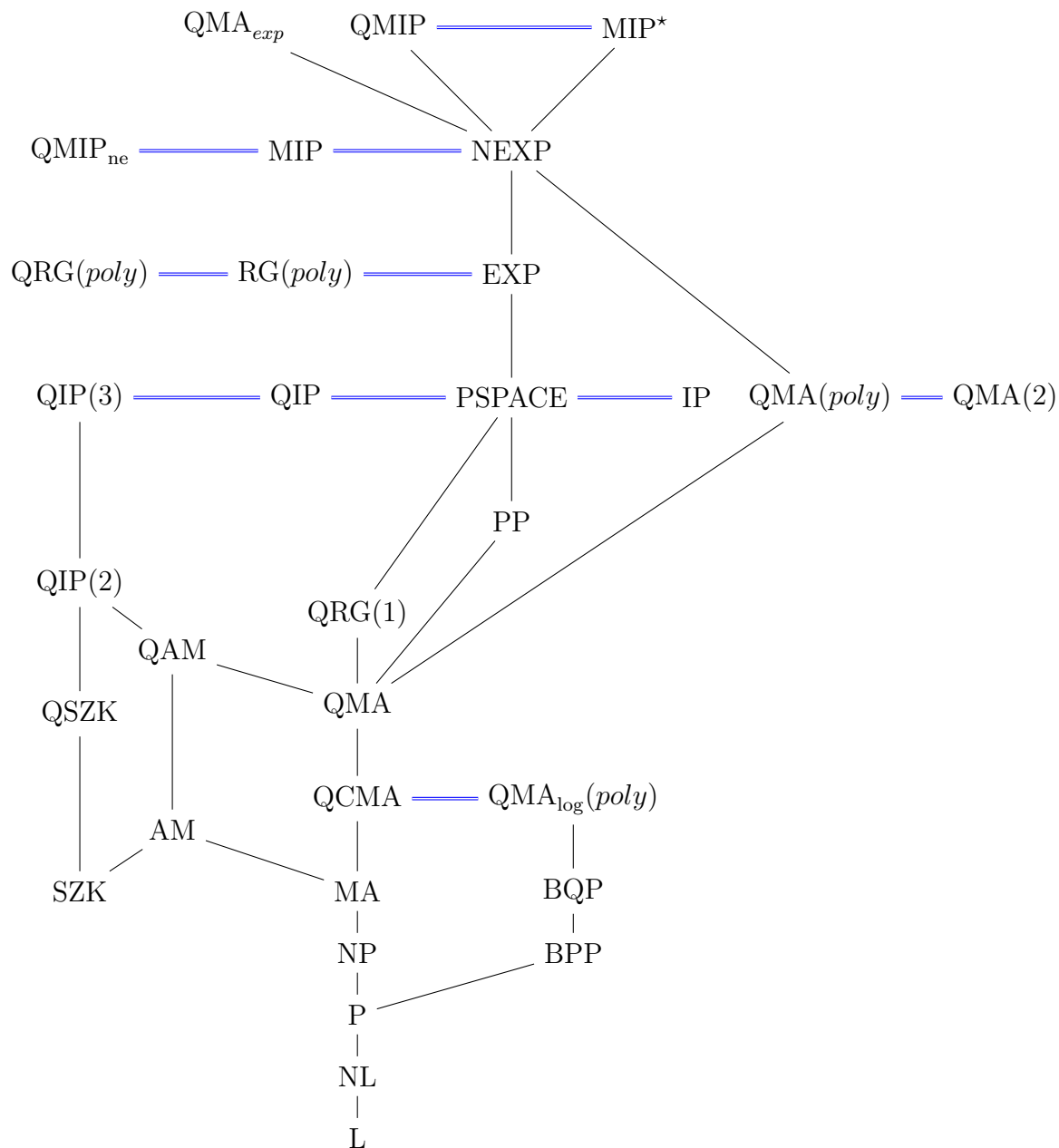


Figure 5.5: A hierarchy with some classical and quantum complexity classes. Horizontal arrows in blue show the equivalence of some classes while vertical arrows show containment in the upper class^a.

^aFor clarity the inclusion arrows for $BPP \subseteq SZK$ and $BQP \subseteq QSZK$ were omitted from the diagram.

```

1 Input: Description of a machine  $M$ ;
2 Run  $M'$  on input  $\langle M, M \rangle$ ;
3 if  $M'$  accepts then
4   |  $D_f(M)$  rejects;
5 end
6 else
7   |  $D_f(M)$  accepts;
8 end

```

Algorithm 4: Time Diagonalization Machine D_f .

most $f(n)$. Next, suppose M' rejected $\langle D_f, D_f \rangle$. This means that D_f does not accept its input in at most $f(2n)$ steps. However, in this case $D_f(D_f)$ accepts in at most $f(2n)$ steps. These contradictions imply that M' does not exist. Since the D_f can be easily simulated in $f(n)^3$ steps [75], we have the following theorem.

Theorem 5.8.1.

$$\text{DTIME}(f(n)) \subset \text{DTIME}(f(n)^3).$$

Using a more elaborated simulation of D_f , it is possible to improve the previous result to show an even tighter separation.

Theorem 5.8.2 (From [75]).

$$\text{DTIME}\left(\frac{f(n)}{\log(n)}\right) \subset \text{DTIME}(f(n)).$$

A very important corollary of this last result is the unconditional separation of P from EXP.

Corollary 5.8.3.

$$\text{P} \neq \text{EXP}.$$

Proof. From the time hierarchy we know that $\text{DTIME}(2^n) \subset \text{DTIME}(2^{3n})$. Moreover, $\text{DTIME}(n^k) \subset \text{DTIME}(2^n)$ for any constant k implying $\text{P} \neq \text{EXP}$. \square

Theorem 5.8.4 (Golden Reduction). *Suppose there is a fixed polynomial $p(n)$, such that there is $p(n)$ time reduction from any language in P to a NP-complete language L . Then, $\text{P} \neq \text{NP}$.*

Proof. Suppose L admits a polynomial time algorithm $q(n)$. Then, any language in P can be decided in time $O(q(p(n)) + p(n))$, via the reduction to L . This result contradicts the time hierarchy, as there are languages in P that require $\omega(q(p(n)) + p(n))$ steps to decide. \square

5.8.2 Space Hierarchy

The space hierarchy proof follows using an analogous language, and a similar diagonalization machine, but it is tighter than the time hierarchy.

$$S_f = \{ \langle M, x \rangle \mid M \text{ accepts } x \text{ using space at most } f(n), n = |\langle M, x \rangle| \}$$

Theorem 5.8.5 (From [75]).

$$\text{DSPACE}(o(f(n))) \subset \text{DSPACE}(f(n)).$$

Chapter 6

Classical PCP

The Probabilistically Checkable Proof (PCP) Theorem is one of the central results in complexity theory. Before diving into the theorem itself, we recall important computational resources that will be traded for others in this result. The class NP contains languages that can be decided in polynomial time by a deterministic Turing Machine that has access to a polynomial sized proof. Without loss of generality, a deterministic machine needs to read the entire proof or else it can be reduced. There are important conjectures stating that the size of the proof needs to be at least of polynomial size for the NP-complete language 3SAT such as the Exponential Time Hypothesis (ETH) [55]. This conjecture is in resonance with the strong conjecture that $P \neq NP$, and the fact that asymptotically smaller proofs lead to asymptotically faster brute force algorithms for 3SAT. Therefore, in general, a deterministic machine needs to inspect the whole polynomial sized proof. If we want a machine to inspect fewer positions, a polynomial time deterministic machine seems incapable of performing this task. It turns out that by using randomness and rewriting the proof, it is possible to circumvent this issue. In the PCP result, the number of inspected positions becomes bounded by a constant. This result alone is not as interesting as m -clause 3SAT instances that are not satisfiable will violate at least $\frac{1}{m}$ fraction of clauses. A simple probabilistic procedure that receives an assignment to this instance's variables as a proof can choose a clause uniformly at random and check if it is satisfiable by inspecting only 3 bits. This verification has an inverse polynomial probability of rejecting an input not in language. What makes the PCP Theorem surprising is that it also guarantees a constant rejection probability while never rejecting inputs in the language (for most PCPs).

The PCP Theorem has the philosophical interpretation that any proof can be rewritten in a way that a probabilistic verification procedure can decide with high probability whether it is correct or not by inspecting only a constant number of its positions. It is important to highlight that this theorem is not a purely theoretical result. It has many ap-

plications to hardness of approximation results since it establishes formal barriers beyond which P would be equal to NP.

In this chapter, we formalize the PCP Theorem using three equivalent formulations which provide different perspectives to this result. We also prove their equivalence. A classical proof can be viewed as a boolean function mapping positions to their contents and some variations of the PCP Theorem exploit this interpretation. For this reason, we introduce some tools used in the analysis of boolean functions. Finally, we state the 3-bit PCP Theorem from Hastad which is the most efficient PCP for binary alphabet, not allowing to condition a query on the result of previous ones.

6.1 PCP as a complexity class

Probabilistic proof verification can be parameterized by the amount of randomness, number of queries, alphabet size of the proof, and completeness and soundness. Each specific parametrization may give rise to a specific complexity class.

Definition 6.1.1. *A language L is in $\text{PCP}_{1,1-\epsilon}(r(n), q(n))_{\Sigma}$ if there exist a probabilistic verifier V and a polynomial p such that:*

- **Completeness:** *for all $x \in L$, then there exists a witness $y \in \Sigma^{p(|x|)}$ such that $P[V(x, y) \text{ accepts}] = 1$.*
- **Soundness:** *for all $x \notin L$ and any $y \in \Sigma^{p(|x|)}$. $P[V(x, y) \text{ accepts}] \leq 1 - \epsilon$.*

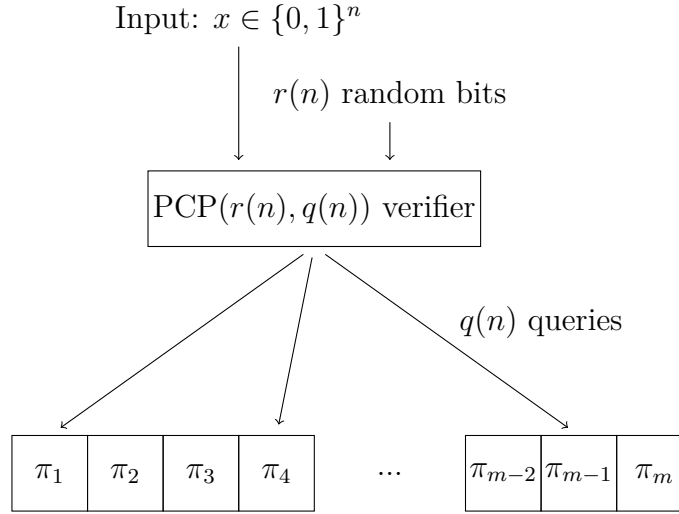
Moreover, V uses at most $r(n)$ random bits and makes at most $q(n)$ queries to y .

A graphical view of the probabilistic proof verification is shown in Figure 6.1.

6.2 PCP Formulations

The classical PCP Theorem allows three equivalent formulations in terms of proof verification, constraint satisfaction problems (CSP), and games. Looking at a problem from a different perspective might foster our comprehension of it, and the PCP Theorem is no exception. In fact, early developments started from a game perspective showing a scaled-up version of the theorem for NEXP.

The proof verification version was the most discussed one so far. The first proof of the PCP Theorem was shown using this version. It can be stated in terms of the class $\text{PCP}(q(n), r(n))$ as follows.

Figure 6.1: PCP verifier inspecting proof π .

Theorem 6.2.1 (PCP Proof Verification). $\text{NP} \subseteq \text{PCP}_{1,1-\epsilon}(\log(n), O(1))_\Sigma$ for constants ϵ and $|\Sigma|$.

Dinur gave a combinatorial proof of the PCP Theorem using the constraint satisfaction problem (CSP) version [29]. Before discussing this version, we establish some notation. The term q -CSP is the restriction of CSPs in which each constraint has arity q , *i.e.* acts on q variables. For an instance φ of a CSP on m clauses, we use the notation $\text{val}(\varphi)$ to denote the fraction of satisfiable clauses. The formal statement of this version is shown next.

Theorem 6.2.2 (PCP CSP). *For every $L \in \text{NP}$ and $n \in \mathbb{N}$, there is a polynomial-time mapping $x \in \{0, 1\}^n \rightarrow \varphi_x$ where φ_x is a q -CSP over $m = \text{poly}(n)$ clauses and $q \in O(1)$, such that:*

- $x \in L \implies \text{val}(\varphi_x) = 1,$
- $x \notin L \implies \text{val}(\varphi_x) \leq 1 - \epsilon,$

for constant ϵ .

The game version is stated as a q -player game, but it could have been equivalently defined using only 2-player games. A q -player game G can be simulated by a 2-player game G' as follows. The referee asks the first player for all the q original questions from G , and at the same time asks one of these q question to the second player, chosen uniformly at random. Without loss of generality, we can assume that the first player's answers would cause the verifier in G to accept. For a given tuple of q questions, if it always

causes the referee to reject, then it can be safely removed and its probability should be subtracted from the value of the new game in order to recover the original value. The q independent strategies in G would cause the referee to reject with probability $1 - \omega(G)$. It means that the first player must lie in a $1 - \omega(G)$ fraction of the cases. Conditioned on the fact that it lied, the answer of the second one will disagree with the first one with probability $\frac{1}{q}$ because the second strategy can not perform better than a concatenation of the q strategies in G . Consequently, the value $\omega(G')$ can be bounded as

$$\omega(G) \leq \omega(G') \leq 1 - \frac{(1 - \omega(G))}{q}.$$

Asymptotically, it is indifferent for the value of the game whether it is a two or a q -player game, for a constant q . The game version is shown next.

Theorem 6.2.3 (PCP Game). *For every $L \in \text{NP}$ and $n \in \mathbb{N}$, there is a polynomial-time mapping $x \in \{0, 1\}^n \rightarrow G_x$ where $G_x = (Q, A, \pi, V)$ is a q -player game with $q \in O(1)$, such that:*

- $x \in L \implies \omega(G_x) = 1$,
- $x \notin L \implies \omega(G_x) \leq 1 - \epsilon$,

for constant ϵ , $|Q| \in O(\text{poly}(n))$, and $|A| \in O(1)$.

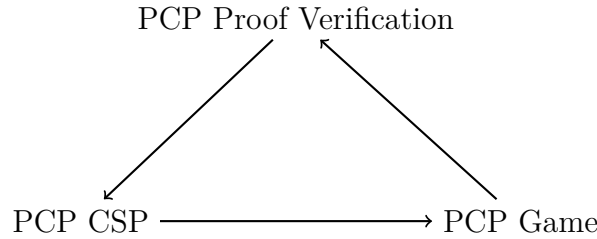


Figure 6.2: Equivalence of different formulations of the PCP Theorem.

(PCP Proof Verification) \implies (PCP CSP). In the proof verification version of the PCP Theorem, the verifier V has access to $r \in O(\log(n))$ random bits that are used to choose q positions to query in the proof. After querying them, it decides to accept or reject the input based on their values. For each r , this decision procedure can be transformed into a polynomial size constraint satisfaction instance φ_r , since V runs in polynomial time. Taking the logical **AND** of all such instances yields the constraint satisfaction problem $\varphi = \bigwedge_{r \in \{0,1\}^{2^{O(\log(n))}}} \varphi_r$ where the variables correspond to positions in the proof. The total number of random strings is polynomial since V uses $O(\log(n))$ random bits making the procedure of building a q -CSP a polynomial time one. It is evident that this transformation preserves the gap implying the PCP CSP Theorem.

(PCP CSP) \implies (PCP Game). A q -CSP φ_x can be transformed into a two player game $G_x = (Q, A, \pi, V)$ in which the referee asks Alice the values of the variables in a uniformly random clause while simultaneously asking Bob the value of a uniformly chosen variable in this clause. Note that the set of questions Q has polynomial size and the set of answers has constant size. If φ_x is satisfiable, then $\omega(G_x) = 1$ provided both players answer according to the same valid assignment. Suppose φ_x is not satisfiable, then $\text{val}(\varphi_x) \leq 1 - \epsilon$. Bob's strategy is just a fixed assignment to variables which, from our hypothesis, violates at least a constant fraction ϵ of the clauses in φ_x . Without loss of generality, we can assume that Alice's answers always satisfies the asked clauses, or otherwise the referee would have readily rejected. Since Bob's assignment violates at least a fraction ϵ of all clauses, there is a fraction ϵ of clauses in which at least the value of one variable Alice and Bob reported does not agree. The referee asks for a variable in a clause uniformly at random causing the rejecting probability to be at least $\frac{\epsilon}{q}$, thus $\omega(G_x) \leq 1 - \frac{\epsilon}{q}$.

(PCP Game) \implies (PCP Proof Verification). For a constant number of players k , asymptotically the value of a game does not change when transformed to a two-player game. For this reason, we assume that the game from the PCP Theorem is of this kind. Fixing a language $L \in \text{NP}$ and an input $x \in \{0, 1\}^n$, we have as hypothesis a game $G_x = (Q, A, \pi, P)$ (to avoid confusion we relabeled the function V in the game definition to P) with $\text{poly}(n)$ questions and a constant number of answers $|A|$. Let S_1 and S_2 be optimum deterministic strategies of the two players. Since the number of questions is polynomial and the answers can be represented by constant size values, these strategies have polynomial size. Their concatenation will be used as the verifier's proof over the alphabet $\Sigma = A$. Moreover, the verifier's queries are specified by the distribution π and the acceptance predicate is given by the function P . A verifier built this way will just simulate the original game without any changes, thus its acceptance probability corresponds to the value of the game $\omega(G_x)$, implying that the original gap is preserved.

6.3 Boolean Functions

Boolean functions are important combinatorial objects of the form $f : \{0, 1\}^n \rightarrow \mathbb{R}$. For instance, a proof π can be viewed as a function mapping positions to their content. Functions may have properties worth checking such as linearity or low degree which are verified in some PCP theorems. To better understand a boolean function it is oftentimes useful to analyze its Fourier representation. This representation provides a simple route to prove important properties about boolean functions [13]. Except when stated otherwise, we work with an alternative representation of boolean values, given by the mapping $b \in \{0, 1\} \rightarrow (-1)^b$. A useful property of this representation is that the logical **XOR** of two

boolean values can be computed by their product. Any boolean function $f : \{1, -1\}^n \rightarrow \mathbb{R}$ can be represented as a multi-linear polynomial. This fact will be clear after inspecting an indicator function for a point $x^0 = x_1^0 \dots x_n^0$ in the hypercube $\{1, -1\}^n$ defined as

$$e_{x^0}(x) = \left(\frac{1 + x_1 x_1^0}{2}\right) \left(\frac{1 + x_2 x_2^0}{2}\right) \dots \left(\frac{1 + x_n x_n^0}{2}\right).$$

This function is 1 if and only if $x = x^0$. Note that the set $\{e_{x^0} : x^0 \in \{1, -1\}^n\}$ forms an orthogonal basis with respect to the inner product

$$\langle f(x), g(x) \rangle = \frac{1}{2^n} \sum_{\alpha \in \{1, -1\}^n} f(\alpha) g(\alpha).$$

In this basis, an arbitrary function f can be written as

$$f(x) = \sum_{x^0 \in \{1, -1\}^n} f(x^0) e_{x^0}(x).$$

We are particularly interested in functions that are linear or, at least, close to linear. For this reason, we briefly review what a linear function is in terms of regular binary values in $\{0, 1\}$, and then move to our new notation. A linear function $f(x) = c \cdot x$ for a constant $c = c_1 \dots c_n$ and $x = x_1 \dots x_n \in \{1, -1\}^n$ can be computed as

$$f(x) = c \cdot x = \sum_{i=1}^n c_i x_i = \sum_{i: c_i \neq 0} x_i.$$

This function is just calculating the parity of a subset of bits $S = \{i : c_i = 1\} \subseteq [n]$ in x . We denote the set S by character, and the linear function in our $\{1, -1\}$ notation becomes a product of the bits in S as

$$f(x) = \chi_S(x) = \prod_{i \in S} x_i.$$

For $S = \emptyset$, we define $\chi_S(x) = 1$. It may not be evident, at first, but the set $\{\chi_S : S \in \{1, -1\}^n\}$ is orthonormal with dimension 2^n . Therefore, it also forms an orthonormal basis for functions $f : \{1, -1\}^n \rightarrow \mathbb{R}$. Note that each χ_S is a linear function or multi-linear monomial. This basis is the famous Fourier basis for boolean functions. To show that the previous set is indeed orthonormal, we calculate the inner product of two arbitrary elements resulting in

$$\begin{aligned}
\langle \chi_S(x), \chi_T(x) \rangle &= \frac{1}{2^n} \sum_{\alpha \in \{1, -1\}^n} \chi_S(\alpha) \chi_T(\alpha) \\
&= \frac{1}{2^n} \sum_{\alpha \in \{1, -1\}^n} \prod_{i \in S} \alpha_i \prod_{j \in T} \alpha_j \\
&= \frac{1}{2^n} \sum_{\alpha \in \{1, -1\}^n} \prod_{i' \in S \cap T} \alpha_{i'}^2 \prod_{j' \in S \Delta T} \alpha_{j'} \\
&= \frac{1}{2^n} \sum_{\alpha \in \{1, -1\}^n} \prod_{j' \in S \Delta T} \alpha_{j'} \\
&= \frac{1}{2^n} \sum_{\alpha \in \{1, -1\}^n} \prod_{k \in U} \alpha_k = \frac{1}{2^n} \sum_{\alpha \in \{1, -1\}^n} \chi_U(\alpha)
\end{aligned}$$

where $U = S \Delta T$. It is clear that if $S = T$, then $U = \emptyset$ and the expression evaluates to 1. If $S \neq T$, then $U \neq \emptyset$ and the expression computes the average parity of uniformly random strings with $|U|$ bits, which is clearly zero.

The parity function, when applied to the component-wise product $\alpha\alpha'$ becomes

$$\chi_S(\alpha\alpha') = \prod_{i \in S} \alpha_i \alpha'_i = \left(\prod_{i \in S} \alpha_i \right) \left(\prod_{i \in S} \alpha'_i \right) = \chi_S(\alpha) \chi_S(\alpha').$$

The Fourier coefficient relative to the character S is given by the expression

$$\hat{f}_S = \langle f(x), \chi_S(x) \rangle = \frac{1}{2^n} \sum_{\alpha \in \{1, -1\}^n} f(\alpha) \chi_S(\alpha).$$

A function f in the Fourier basis is then

$$f(x) = \sum_{S \subseteq [n]} \hat{f}_S \chi_S(x).$$

Since the Fourier basis forms an orthonormal basis for the functional space of boolean functions, we have that any boolean function can be uniquely written as a multi-linear polynomial. As we are working with boolean values as input, even powers of bits can be replaced by 1, whereas odd powers can be replaced by the value of the bit raised to 1. Therefore, there is no need for anything other than a multi-linear representation.

For functions $f : \{1, -1\}^n \rightarrow \{1, -1\}$, the following holds

$$\langle f(x), f(x) \rangle = \frac{1}{2^n} \sum_{\alpha \in \{1, -1\}^n} f(\alpha)^2 = 1.$$

If we compute the previous inner product in terms of the Fourier representation of f we have

$$\begin{aligned}
1 = \langle f(x), f(x) \rangle &= \frac{1}{2^n} \sum_{\alpha \in \{0,1\}^n} \sum_{S \subseteq [n]} \sum_{T \subseteq [n]} \hat{f}_S \chi_S(\alpha) \hat{f}_T \chi_T(\alpha) \\
&= \sum_{S \subseteq [n]} \sum_{T \subseteq [n]} \hat{f}_S \hat{f}_T \left(\frac{1}{2^n} \sum_{\alpha \in \{0,1\}^n} \chi_S(\alpha) \chi_T(\alpha) \right) \\
&= \sum_{S \subseteq [n]} \hat{f}_S^2.
\end{aligned}$$

The identity $\sum_{S \subseteq [n]} \hat{f}_S^2 = 1$ is known as the Parseval's identity.

Testing if f is exactly linear would require querying its value on all inputs. Nevertheless, the next simple test can enforce that it is “almost” linear.

- 1 Choose $\alpha, \alpha' \in \{0,1\}^n$ at random;
- 2 Query $f(\alpha), f(\alpha'), f(\alpha + \alpha')$;
- 3 Accept iff $f(\alpha + \alpha') = f(\alpha) + f(\alpha')$;

Algorithm 5: BLR Linearity Test.

The next theorem makes precise the notion of “almost linear”. If the linearity test succeeds with high probability, f is a linear function for a large fraction of the input.

Theorem 6.3.1 (BLR). *The BLR linearity test satisfies two conditions.*

- (i) If f is linear, then $\Pr[f \text{ passes BLR test}] = 1$.
- (ii) Suppose $\Pr[f \text{ passes BLR test}] \geq 1 - \epsilon$ for some $\epsilon > 0$, then there is a coefficient c such that $f(\alpha) = c\alpha$ for $1 - 2\epsilon$ fraction of $\alpha \in \{0,1\}^n$.

Proof. We use the alternative binary representation $\{1, -1\}$ for f . Item (i) is straightforward, and follows from the linearity of f . For item (ii), first note that

$$\begin{aligned}
\Pr[f \text{ passes BLR test}] &= \Pr[f(\alpha)f(\alpha') = f(\alpha + \alpha')] \\
&= \mathbf{E}[f(\alpha)f(\alpha')f(\alpha + \alpha')] + \Pr[f(\alpha)f(\alpha') \neq f(\alpha + \alpha')].
\end{aligned}$$

Rearranging the terms and using the lower bound on the BLR acceptance, the expectation becomes

$$\begin{aligned}
\mathbf{E}[f(\alpha)f(\alpha')f(\alpha + \alpha')] &= \Pr[f \text{ passes BLR test}] - \Pr[f(\alpha)f(\alpha') \neq f(\alpha + \alpha')] \\
&\geq 1 - \epsilon - \epsilon = 1 - 2\epsilon.
\end{aligned}$$

Expanding the expectation, we have

$$\mathbf{E}[f(\alpha)f(\alpha')f(\alpha + \alpha')] = \sum_{\alpha \in \{0,1\}^n} \sum_{\alpha' \in \{0,1\}^n} \frac{1}{2^{2n}} f(\alpha)f(\alpha')f(\alpha + \alpha').$$

Now, we use the Fourier representation of f , resulting in

$$\begin{aligned}
\mathbf{E}[f(\alpha)f(\alpha')f(\alpha\alpha')] &= \frac{1}{2^{2n}} \sum_{\alpha \in \{0,1\}^n} \sum_{\alpha' \in \{0,1\}^n} \sum_{S,U,T \subseteq [n]} \hat{f}_S \chi(\alpha) \hat{f}_U \chi(\alpha') \hat{f}_T \chi(\alpha\alpha') \\
&= \frac{1}{2^{2n}} \sum_{\alpha \in \{0,1\}^n} \sum_{\alpha' \in \{0,1\}^n} \sum_{S,U,T \subseteq [n]} \hat{f}_S \hat{f}_U \hat{f}_T \chi_S(\alpha) \chi_U(\alpha') \chi_T(\alpha) \chi(\alpha') \\
&= \frac{1}{2^{2n}} \sum_{\alpha \in \{0,1\}^n} \sum_{\alpha' \in \{0,1\}^n} \sum_{S,U,T \subseteq [n]} \hat{f}_S \hat{f}_U \hat{f}_T \chi_S(\alpha) \chi_U(\alpha') \chi_T(\alpha) \chi_T(\alpha') \\
&= \sum_{S,U,T \subseteq [n]} \hat{f}_S \hat{f}_U \hat{f}_T \left(\frac{1}{2^n} \sum_{\alpha \in \{0,1\}^n} \chi_S(\alpha) \chi_T(\alpha) \right) \left(\frac{1}{2^n} \sum_{\alpha' \in \{0,1\}^n} \chi_U(\alpha') \chi_T(\alpha') \right).
\end{aligned}$$

Recall that the character functions form an orthonormal basis. The expectation can be simplified as

$$\begin{aligned}
\mathbf{E}[f(\alpha)f(\alpha')f(\alpha\alpha')] &= \sum_{S,U,T \subseteq [n]} \hat{f}_S \hat{f}_U \hat{f}_T \langle \chi_S(x), \chi_T(x) \rangle \langle \chi_U(x), \chi_T(x) \rangle \\
&= \sum_{S \subseteq [n]} \hat{f}_S^3 \quad \text{(Orthonormality)} \\
&\leq \max_{U \subseteq [n]} \hat{f}_U \sum_{S \subseteq [n]} \hat{f}_S^2 \\
&= \max_{U \subseteq [n]} \hat{f}_U. \quad \text{(Parseval)}
\end{aligned}$$

If the function f passes the BLR test with probability at least $1 - \epsilon$, we have the following lower bound for the largest Fourier coefficient

$$\max_{U \subseteq [n]} \hat{f}_U \geq \mathbf{E}[f(\alpha)f(\alpha')f(\alpha\alpha')] \geq 1 - 2\epsilon.$$

□

6.4 Hastad 3-bit PCP

Hastad's 3-bit Theorem is one of the most query efficient PCP theorems for the proof verification variant. It states that any language in NP can be probabilistically verified by reading only 3 bits in a polynomial size binary proof. This section is based on the results of [10]. More formally, this theorem can be presented as follows.

Theorem 6.4.1 (Hastad's 3-bit PCP [10]). *For every $\delta > 0$ and every language $L \in \text{NP}$, there is a PCP verifier V for L making three (binary) queries having completeness parameter $1 - \delta$ and soundness parameter at most $\frac{1}{2} + \delta$.*

Moreover, the tests used by V are linear. That is, given a proof $\pi \in \{0, 1\}^m$, V chooses a triple $(i_1, i_2, i_3) \in [m]^3$ and $b \in \{0, 1\}$ according to some distribution, and accepts if and only if $\pi_{i_1} + \pi_{i_2} + \pi_{i_3} = b \pmod{2}$.

We can see the proof π as given by a prover trying to maximize the number of satisfied equations in a system of linear equations over $\text{GF}(2)$. This problem is known as MAX- k -XOR, in which k is the maximum number of variables per equation. In this case, k is simply three. Contrary to several other PCP results, the verifier does not have perfect completeness and this is indeed important in this case. Checking if a system of linear equations can be simultaneously satisfied is in P.

6.4.1 Hardness of Approximation for 3SAT

PCP theorems are notorious for their application in the hardness of approximation for many problems of practical use. In the case of Theorem 6.4.1, it provides a hardness of approximation for the optimization version of 3SAT denoted MAX-3-SAT. As mentioned earlier, there is a MAX-3-XOR instance underlying the verification procedure. Unless $P = NP$, it is impossible to approximate the number of satisfiable equations of MAX-3-XOR within a $\frac{1}{2} + \delta$ factor for any $\delta > 0$. Given this hardness result for MAX-3-XOR, it suffices to give a reduction to 3SAT, which is still gapped. We have two types of equations, namely, $a + b + c = 0$ and $a + b + c = 1$. Each can be replaced by a 3CNF formula as follows:

$$\begin{aligned} a + b + c = 0 &\rightarrow (\neg a \vee b \vee c) \wedge (a \vee \neg b \vee c) \wedge (a \vee b \vee \neg c) \wedge (\neg a \vee \neg b \vee \neg c), \text{ and} \\ a + b + c = 1 &\rightarrow (\neg a \vee \neg b \vee c) \wedge (\neg a \vee b \vee \neg c) \wedge (a \vee \neg b \vee \neg c). \end{aligned}$$

Since each equation gives rise to at most four clauses and at least one of them is violated if the equation is violated, it is not possible to satisfy a fraction $1 - \frac{1}{4}(\frac{1}{2} - \delta)$ of clauses, unless $P = NP$. We formalize this result in the next corollary.

Corollary 6.4.2. *It is NP-hard to determine whether a fraction $\frac{7}{8} + \epsilon$ of the clauses in a MAX-3-SAT instance is satisfiable or not, for every $\epsilon > 0$.*

6.4.2 Previous Result

A $\text{GAP2CSP}_W(\epsilon)$ is a constraint satisfaction problem over an alphabet W and in which each constraint acts on two variables. Moreover, it has an additional gap property. If an instance of this problem is not satisfiable, then at most an ϵ fraction of constraints may be simultaneously satisfied. Raz showed that is possible to achieve an arbitrarily small ϵ at the cost of an alphabet blow up as captured by the next Theorem.

Theorem 6.4.3 (Raz [10]). *There is a $c > 1$ such that for every $t > 1$, $\text{GAP-2-CSP}_W(\varepsilon)$ is NP-hard for $\varepsilon = 2^{-t}$, $W = 2^{ct}$, and this is true also for 2CSP instances that are regular and have the projection property.*

The projection property for a 2CSP means that for each constraint $\varphi_r(i, j)$ acting on variables i and j if we know the value of variable i , then there is only one value of variable j that satisfies this constraint. Therefore, with this property, each constraint can be equivalently represented by a function $h : [W] \rightarrow [W]$ from the first variable to the second. For every $u \in [W]$, we define the set $h^{-1}(u) = \{w \in [W] | h(w) = u\}$. Note that these sets form a partition of $[W]$. The next figure illustrates these objects.

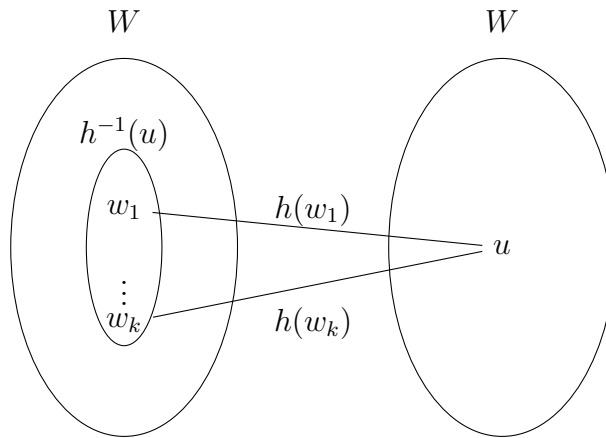


Figure 6.3: Function $h : [W] \rightarrow [W]$ corresponding to a projection constraint.

The Raz's Theorem 6.4.3 is used as the starting point for the Hastå 3-bit PCP. Note that Raz's result already implies that $\text{NP} \in \text{PCP}(O(\log(n)), 2)$ via a protocol that random selects a clause, and then reads its associated variables. The issue with this approach is that each variable assumes values in an alphabet of size 2^{ct} . Hastå's result improve's upon this simple protocol as it reads only three bits, in total.

6.4.3 Long Code

An arbitrary number w in the set $[W]$ can be encoded in a plethora of ways. One possible approach would be to use a binary representation of w that has $O(\log(|W|))$ bits. Alternatively, it is possible to construct a coordinate (or dictatorship) function $f(x_1, \dots, x_{|W|}) = x_w$ and store a binary table in $\{\pm 1\}^{2^{|W|}}$. This table is the concatenation of the outputs corresponding to all possible inputs when the latter are sorted in ascending order. This table represent the long code of w . The designation long is evident since the long code is doubly exponential compared to the binary representation. The expensive

encoding plays an import role in Hastå's PCP as it is used to represent the values of each variable in an assignment to a $\text{GAP-2-CSP}_W(\varepsilon)$ instance.

6.4.4 PCP Protocol

The PCP verifier protocol checks whether an instance φ of $\text{GAP-2-CSP}_W(\varepsilon)$ is satisfiable or not. Let m be the number of clauses and n the number of variables in φ . This verifier expects an assignment to each variable in the long code representation leading to a proof of length $n2^{|W|}$.

If φ is not satisfiable, Raz's Theorem 6.4.3 ensures that at most a fraction ε of the clauses is satisfiable. The verifier simply chooses a clause $\varphi_r(i, j)$ at random and takes advantage of the long code representation to check that $\text{val}[i] = w$ and $\text{val}[j] = u = h(w)$, using only 3 queries, where $h : [W] \rightarrow [W]$ is the function representing this projection constraint.

Before we formally describe the protocol, we define a function $\mathcal{H}^{-1}(y) : \{\pm 1\}^{|W|} \rightarrow \{\pm 1\}^{|W|}$ such that $\mathcal{H}^{-1}(y)_w = y_{h(w)}$. Each position u in the original string y is mapped to each position in $h^{-1}(u)$.

We denote the Hastå's PCP verifier by V_H . A precise description of it is given next.

Input: a $\text{GAP-2-CSP}_W(\varepsilon)$ instance φ and an assignment using the long code encoding to each variable.

- 1) Choose a clause $\varphi_r = (i, j)$ uniformly at random where i and j denote its associated variables and f and g their respective long codes ;
- 2) Choose $x, y \in_R \{\pm 1\}^W$ and $z \in \{\pm 1\}^W$ such that $z_i = 1$ with probability $1 - \rho$ and $z_i = -1$ with probability ρ ;
- 3) Accept iff $f(x)g(y) = f(\mathcal{H}^{-1}(y)xz)$;

Algorithm 6: Hastå's PCP verifier V_H .

Lemma 6.4.4 (Main (from [10])). *If φ is satisfiable, then there is a proof which V_H accepts with probability $1 - \rho$. If $\text{val}(\varphi) \leq \varepsilon$, then V_h accepts no proof with probability more than $\frac{1}{2} + \delta$, where $\delta = \sqrt{\frac{\varepsilon}{\rho}}$.*

The completeness part is straightforward as shown by the next claim.

Claim 6.4.5. *The Basic Hastå Test has completeness probability $1 - \rho$.*

Proof. We can assume that the proof π contains the long code representation of each variable corresponding to a valid assignment to φ . Denote by w and u the value encoded by functions f and g , respectively, of the chosen constraint φ_r . Since we have a valid assignment, it holds that $h(w) = u$. The verifier accepts if and only if $f(x)g(y) = f(\mathcal{H}^{-1}(y)xz)$ or equivalently $f(x)g(y)f(\mathcal{H}^{-1}(y)xz) = 1$. Simplifying the previous expression, we have

$$\begin{aligned}
1 &= f(x)g(y)f(\mathcal{H}^{-1}(y)xz) = x_w y_u \mathcal{H}^{-1}(y)_w x_w z_w \\
&= x_w y_u y_{h(w)} x_w z_w \\
&= x_w^2 y_u^2 z_w = z_w.
\end{aligned}$$

By construction, we know that the probability of $z_w = 1$ is $1 - \rho$, completing the proof. \square

Soundness requires that we introduce more definitions and an auxiliary lemma. We define the set $h_2(\alpha)$ as follows

$$h_2(\alpha) = \{u \in [W] \mid |h^{-1}(u) \cap \alpha| \text{ is odd}\}.$$

This definition might seem arbitrary now, but requiring that the intersection be odd will always provide a variable one variable to cancel another latter on (remember that $x^2 = 1$ for $x \in \{\pm 1\}$).

A function f is said to be bifoldded if $f(-x) = -f(x)$ (note that this is the usual odd property, but such different terminology is more common in this context). Without loss of generality, it is possible to assume that the assignment to each variable is given by a bifoldded function. It is enough to ignore half of the long code and query the same position for $f(x)$ and $f(-x)$, multiplying the result by -1 in the latter case.

With these new definitions, we are ready to present the auxiliary lemma.

Lemma 6.4.6 (Adapted from [10]). *Let $f, g : \{\pm 1\}^{|W|} \rightarrow \{\pm 1\}$, be bifoldded functions and $h : [W] \rightarrow [W]$ be such that they pass the Basic Hastad Test 6 with probability at least $\frac{1}{2} + \delta$. Then*

$$\sum_{\alpha \subseteq [W], \alpha \neq \emptyset} \hat{f}_\alpha^2 \hat{g}_{h_2(\alpha)} \frac{1}{\sqrt{|\alpha|}} \geq \delta \sqrt{\rho}. \quad (6.1)$$

Proof. Since f and g pass the Basic Hastad Test with probability with probability at least $\frac{1}{2} + \delta$, we have

$$2\delta \leq \mathbf{E}_{x,y,z}[f(x)g(y)f(\mathcal{H}^{-1}(y)xz)].$$

Expanding f and g in the Fourier basis results in

$$\begin{aligned}
2\delta &\leq \mathbf{E}_{x,y,z}[f(x)g(y)f(\mathcal{H}^{-1}(y)xz)] \\
&= \mathbf{E}_{x,y,z}\left[\left(\sum_{\alpha} \hat{f}_\alpha \chi_\alpha(x)\right)\left(\sum_{\beta} \hat{g}_\beta \chi_\beta(y)\right)\left(\sum_{\gamma} \hat{f}_\gamma \chi_\gamma(\mathcal{H}^{-1}(y)xz)\right)\right] \\
&= \sum_{\alpha} \sum_{\beta} \sum_{\gamma} \hat{f}_\alpha \hat{g}_\beta \hat{f}_\gamma \mathbf{E}_{x,y,z}[\chi_\alpha(x) \chi_\gamma(x) \chi_\beta(y) \chi_\gamma(\mathcal{H}^{-1}(y)) \chi_\gamma(z)] \text{ (Linearity of expectation)} \\
&= \sum_{\alpha} \sum_{\beta} \hat{f}_\alpha^2 \hat{g}_\beta \mathbf{E}_{y,z}[\chi_\beta(y) \chi_\alpha(\mathcal{H}^{-1}(y)) \chi_\alpha(z)] \text{ (Orthogonality of } \chi_\alpha).
\end{aligned}$$

The random variables z and y are independent thus it is possible to split the expectation in a term that depends only on z as

$$\begin{aligned} \mathbf{E}[\chi_\alpha(z)] &= \mathbf{E}\left[\prod_{w \in \alpha} z_w\right] \\ &= \prod_{w \in \alpha} \mathbf{E}[z_w] \\ &= \prod_{w \in \alpha} \mathbf{E}[z_1] = (1 - 2\rho)^{|\alpha|}, \end{aligned}$$

and a term that depends only on y . This leads to the following simplification

$$\begin{aligned} &= \sum_{\alpha} \sum_{\beta} \hat{f}_{\alpha}^2 (1 - 2\rho)^{|\alpha|} \hat{g}_{\beta} \mathbf{E}_y[\chi_{\beta}(y) \chi_{\alpha}(\mathcal{H}^{-1}(y))] \\ &= \sum_{\alpha} \sum_{\beta} \hat{f}_{\alpha}^2 (1 - 2\rho)^{|\alpha|} \hat{g}_{\beta} \mathbf{E}_y\left[\prod_{u \in \beta} y_u \prod_{w \in \alpha} y_{h(w)}\right]. \end{aligned}$$

Note that the previous expectation over y is one if the term $\prod_{w \in \alpha} y_{h(w)}$ contains an odd number of each y_u , and it is zero otherwise. Rephrasing this condition in terms of the notation h_2 , the expectation is one if $\beta = h_2(\alpha)$, and zero otherwise.

$$= \sum_{\alpha \neq \emptyset} \hat{f}_{\alpha}^2 (1 - 2\rho)^{|\alpha|} \hat{g}_{h_2(\alpha)}.$$

Since f and g are bifoldded, \hat{f}_{\emptyset} and \hat{g}_{\emptyset} are both equal to zero. Using the inequality $(1 - 2\rho)^{|\alpha|} \leq \frac{2}{\sqrt{\rho^{|\alpha|}}}$, we complete the proof. \square

Lemma 6.4.7 (From [10]). *Suppose φ is an instance of a 2CSP_W such that $\text{val}(\varphi) < \varepsilon$. If ρ and δ satisfy $\rho\delta^2 > \varepsilon$, then verifier V_H accepts any proof with probability at most $\frac{1}{2} + \delta$.*

Proof. We show that if the verifier accepts with probability more than $\frac{1}{2} + \delta$, then it is possible to construct an assignment that satisfies at least $\rho\delta^2$ fraction of the constraints, contradicting the assumption $\text{val}(\phi) < \varepsilon$. This assignment is not built explicitly. Instead we show that a probabilistic assignment satisfies, in expectation, at least $\rho\delta^2$ fraction of constraints. Therefore, by the probabilistic method, there is one assignment at least as good as the expectation. We need to show that the following holds,

$$\mathbf{E}_{\pi}[\mathbf{E}_{r \in [m]}[\pi \text{ satisfies the constraint } \varphi_r(i, j)]] \geq \rho\delta^2. \quad (6.2)$$

Denote by π' the proof that makes V_H accept with probability at least $\frac{1}{2} + \delta$. We use this proof to construct our probabilistic proof π . In π' , we can treat each assignment to a variable $i \in [W]$ as a bifoldded function $f_i : \{\pm 1\}^{|W|} \rightarrow \{\pm 1\}$. Each function f_i gives rise to a distribution \mathcal{D}_i over $[W]$, from which the variables of π will be drawn. The Fourier coefficients also play an important role in these distributions as shown in the following procedure.

Input: A function $f_i : \{\pm 1\}^{|W|} \rightarrow \{\pm 1\}$ with Fourier coefficients \hat{f}_α
 1) Select set $\alpha \subseteq [W]$ with probability \hat{f}_α^2 ;
 2) Select $w \in \alpha$ with probability $\frac{1}{|\alpha|}$;

Algorithm 7: Draw a value $w \in [W]$ according to the i^{th} variable distribution \mathcal{D}_i .

Since f_i is bifoldded, the coefficient \hat{f}_\emptyset is zero. We also have $\sum_\alpha \hat{f}_\alpha^2 = 1$. Consequently, we have a well defined distribution.

The probability that π' satisfies φ_r given that V_H chose constraint r is denoted by

$$\Pr[\pi' \text{ satisfies } \varphi_r | r] \geq \frac{1}{2} + \delta_r,$$

where δ_r can be negative. The expectation $\mathbf{E}_r[\delta_r] = \delta$ is confirmed by the computation

$$\begin{aligned} \frac{1}{2} + \delta &= \Pr[V_H \text{ accepts } \pi'] \\ &= \mathbf{E}_r\left[\frac{1}{2} + \delta_r\right] \\ &= \frac{1}{2} + \mathbf{E}_r[\delta_r]. \end{aligned}$$

If we show that

$$\Pr_\pi[\pi \text{ satisfies } \varphi_r] \geq \rho \delta_r^2, \tag{6.3}$$

then the expectation 6.2 reduces to $\rho \mathbf{E}_r[\delta_r^2]$, due to linearity. By Jensen's inequality ($\mathbf{E}[f(X)] \geq f(\mathbf{E}[X])$ for a convex function f), we have

$$\rho \mathbf{E}_r[\delta_r^2] \geq \rho (\mathbf{E}_r[\delta_r])^2 = \rho \delta^2.$$

Therefore, it is sufficient to show 6.3. Given a constraint $\varphi_r(i, j)$, we need to find the probability that a random assignment $w \sim \mathcal{D}_i$ and $u \sim \mathcal{D}_j$ (the symbol \sim stands for: drawn from) satisfies $h(w) = u$ where $h : [W] \rightarrow [W]$. We introduce the indicator function I_r that is one if and only if this condition is met. When drawing from \mathcal{D}_i , suppose we selected the set α . In this case, when drawing from \mathcal{D}_j , if we chose the set $\beta = h_2(\alpha)$,

we know that no matter which $u \in h_2(\alpha)$ we pick, there will always exist a $w \in \alpha$ such that $h(w) = u$. This happens because the intersection $h^{-1}(u) \cap \alpha$ is never empty, by the definition of $h_2(\alpha)$. The probability of selecting the appropriate w from α is at least $\frac{1}{|\alpha|}$. So, we have a probability $\frac{1}{|\alpha|} \hat{f}_\alpha^2 \hat{g}_{h_2(\alpha)}^2$ for this event. We can sum over all sets α to establish a lower bound on $\mathbf{E}_{\mathcal{D}_i, \mathcal{D}_j}[I_r]$ as

$$\sum_{\alpha} \frac{1}{|\alpha|} \hat{f}_\alpha^2 \hat{g}_{h_2(\alpha)}^2 \leq \mathbf{E}_{\mathcal{D}_i, \mathcal{D}_j}[I_r]. \quad (6.4)$$

Now, we need to make some adjustments to the inequality of Lemma 6.4.6 to derive the final lower bound. Remember that this auxiliary Lemma, for constraint r , states that

$$2\delta_r \sqrt{\rho} \leq \sum_{\alpha} \hat{f}_\alpha^2 \hat{g}_{h_2(\alpha)} \frac{2}{\sqrt{|\alpha|}}.$$

We apply the Cauchy-Schwarz inequality with $\hat{f}_\alpha \hat{g}_{h_2(\alpha)} \frac{2}{\sqrt{|\alpha|}}$ as the components of the first vector, and \hat{f}_α as the components of the second one. This results in

$$\begin{aligned} 2\delta_r \sqrt{\rho} &\leq \sum_{\alpha} \hat{f}_\alpha^2 \hat{g}_{h_2(\alpha)} \frac{2}{\sqrt{|\alpha|}} \\ &\leq \sqrt{\sum_{\alpha} \frac{1}{|\alpha|} \hat{f}_\alpha^2 \hat{g}_{h_2(\alpha)}^2} \sqrt{\sum_{\alpha} \hat{f}_\alpha^2}. \end{aligned}$$

Using the normalization condition $\sum_{\alpha} \hat{f}_\alpha^2 = 1$, and substituting this previous inequality in 6.4, we conclude the proof as we can now write

$$\delta_r^2 \rho \leq \mathbf{E}_{\mathcal{D}_i, \mathcal{D}_j}[I_r] = \Pr_{\pi}[\pi \text{ satisfies } \varphi_r].$$

□

Chapter 7

Variations of Unentangled Provers

The question of tight bounds for $\text{QMA}(k)$ has been open for more than a decade, and it became an important open problem in quantum complexity [65]. The best known lower bound is QMA , as the verifier can ignore additional proofs. While the best known upper bound is NEXP , as it is enough to non-deterministically guess the classical description of $k \in \text{poly}$ polynomial qubits states and then compute the acceptance probability of the quantum verifier. These bounds are just the trivial ones, not being specially helpful for a thorough understanding of $\text{QMA}(k)$. To have an idea of the lack of knowledge about unentangled proofs in term of classical classes, first recall that the class QMA is contained in PP . Therefore, the precise power of unentangled proofs may range from a gap potentially larger than PP to NEXP . In this context, studying variations of $\text{QMA}(k)$ that allow different resources, or use different constraints, may shed light on its capabilities and limitations. Delineating the boundaries on which extra resources let the class unchanged will give more flexibility in the design of new protocols. Furthermore, understanding what resources make this class seemingly bigger than its known lower bound may give clues as to what resources we may want to simulate in the original class.

7.1 Alternative characterization of $\text{QMA}(\text{poly})_{\log(n)}$

A variation of $\text{QMA}(k)$ that has attracted much attention is its restriction to logarithmic size proofs. Any unitary acting on a logarithmic number of qubits admits a polynomial size classical description with respect to a fixed universal gate set. This observation leads to an alternative characterization of $\text{QMA}(\text{poly})_{\log(n)}$ as we show next. First, we state the following corollary, implied by the Solovay-Kitaev Theorem.

Corollary 7.1.1 (From [76]). *For any unitary operator U on l qubits and $\epsilon > 0$, there exists a circuit $C_{U,\epsilon}$ such that it is made up of $O(5^l \log^3(\frac{5^l}{\epsilon}))$ gates from the $\{\mathbf{H}, \mathbf{T}, \mathbf{CNOT}\}$*

gate set, and for all $|\psi\rangle \in \mathbb{C}^{2^l}$ it holds that

$$\frac{1}{2} \left\| U|\psi\rangle\langle\psi|U^\dagger - |\xi\rangle\langle\xi| \right\|_1 \leq \epsilon,$$

where $|\xi\rangle = C_{U,\epsilon}(|\psi\rangle)$.

One consequence of this corollary is that any unitary acting on a logarithmic number of qubits can be efficiently implemented up to an exponentially small error, resulting in the following lemma. It is rather simple and it was discovered independently by the author who latter found this result in [43]. Their result is based on the fact that it is possible to have an efficient classical description of logarithmic size quantum state.

Lemma 7.1.2. $\text{QCMA} = \text{QMA}(\text{poly})_{\log(n)}$.

Proof. The direction $\text{QCMA} \subseteq \text{QMA}(\text{poly})_{\log(n)}$ is trivial. Let y be the classical witness and k its size in the QCMA protocol. Then it is possible to create a $\text{QMA}(k)_{\log(n)}$ protocol in which the i^{th} prover sends the i^{th} bit of y . It suffices to concatenate these bits and run the original verifier. The Lemma 7.1.3 shows the reverse containment. \square

Lemma 7.1.3. $\text{QMA}(k)_{\log(n)} \subseteq \text{QCMA}$ for $k \in \text{poly}$.

Proof. For each $i \in [k]$, let $|\psi_i\rangle$ be the i^{th} witness in $|\psi\rangle = |\psi_1\rangle \otimes \cdots \otimes |\psi_k\rangle$ and let U_i be a unitary that takes $|0\rangle$ to $|\psi_i\rangle$. Note that since this unitary acts on a logarithmic number of qubits, an efficient implementation to reach an exponentially low error can be done in polynomial time 7.1.1. Moreover, this implementation has also an efficient classical description.

The QCMA prover may send these k classical descriptions. The verifier performs the following action:

$$(U_1 \otimes \cdots \otimes U_k)|0\rangle_1 \otimes \cdots \otimes |0\rangle_k,$$

obtaining $|\psi'\rangle = |\psi'_1\rangle \otimes \cdots \otimes |\psi'_k\rangle$ that is exponentially close to $|\psi\rangle$, in the trace distance. \square

Using the equivalence $\text{QMA} = \text{QMA}(\text{poly})_{1-\text{LOCC}}$ and the trivial containment $\text{QCMA} \subseteq \text{QMA}$, we have the next corollary.

Corollary 7.1.4. $\text{QMA}(\text{poly})_{\log(n)} = \text{QCMA} \subseteq \text{QMA} = \text{QMA}(\text{poly})_{1-\text{LOCC}}$.

It gives an alternative route to attempt the proof $\text{QCMA} \neq \text{QMA}$.

7.2 Perfect Completeness of $\text{QMA}(\text{poly})_{\log(n)}$

With the equality $\text{QMA}(\text{poly})_{\log(n)} = \text{QCMA}$, it is possible to show that $\text{QMA}(\text{poly})_{\log(n)}$ remains the same even with the additional constraint of having perfect completeness. For this, we make use of the next theorem.

Theorem 7.2.1 (From [60]). $\text{QCMA} = \text{QCMA}_1$.

Lemma 7.2.2. $\text{QMA}(\text{poly})_{\log(n)} = \text{QMA}(\text{poly}, 1, \frac{1}{3})_{\log(n)}$.

Proof. Combining Lemma 7.1.3 and Theorem 7.2.1, we have the following sequence of inclusions $\text{QMA}(\text{poly})_{\log(n)} \subseteq \text{QCMA} \subseteq \text{QCMA}_1$. For every language L in $\text{QMA}(\text{poly})_{\log(n)}$ and $x \in \{0, 1\}^*$, there is a QCMA verifier V_x that has perfect completeness and constant soundness. We construct a $\text{QMA}(\text{poly})_{\log(n)}$ verifier V'_x for this language that has perfect completeness. For x in L , let y be a certificate that causes V_x to accept with probability 1. In this case, the verifier V'_x expects that each prover sends one of the polynomially many bits in y . It measures them in the computational basis and apply V_x . If $x \notin L$, no matter what bits were received, the soundness when applying V_x remains the same. \square

7.3 Variations of Unentangled Multi-Prover QIP

The class $\text{QMIP}_{ne}(k)$ consists of languages that can be verified by a multi-prover quantum interactive proof systems of k unentangled provers. This is the quantum analogue of MIP in which provers do not share entanglement. We are interested in several variations of this quantum class concerning the type of messages (i.e. classical or quantum), their size, the number of provers, and the completeness and soundness parameters. For this reason, we establish a syntax to express these variations.

The symbols c , q , and e are used to denote classical, quantum, and EPR halves messages, respectively. A sequence $\hat{s} = [t_1 t_2 \dots t_k]_i$ with $t_j \in \{c, q, e\}$ for $j \in [k]$ denotes an interaction of the verifier with the i^{th} prover. If k is odd, the prover initiates the first message. Otherwise, it is the verifier that starts by sending a query to the prover. Two sequences \hat{s}_1 and \hat{s}_2 can be combined in two ways, namely, $\hat{s}_1 \rightarrow \hat{s}_2$ and $\hat{s}_1 \parallel \hat{s}_2$. In the first case, interaction \hat{s}_1 happens before interaction \hat{s}_2 , while in the second they happen in parallel. A parallel interaction is only well defined when the sets of provers involved in \hat{s}_1 and \hat{s}_2 are disjoint and the interactions are of the form $[t_1 t_2]_i$ or $[t_1]_i$. The parallel interaction prevents the questions from verifier to depend on the states received in that round. Moreover, k sequential repetitions of a sequence \hat{s} are represented as $\hat{s}^k = \hat{s} \rightarrow \dots \rightarrow \hat{s}$.

To specify a restriction of $\text{QMIP}_{ne}(k)$ to the sequence of interactions in \hat{s} , we prefix this class with it as in $\hat{s}\text{-QMIP}_{ne}(k)$. For instance, $\text{QMIP}_{ne}(k) = ([qq]_1 \parallel \cdots \parallel [qq]_k)^{poly} - \text{QMIP}_{ne}(k)$.

It is also possible to restrict the maximum length of each message in terms of qubits for quantum messages, and in bits for classical ones. Let $f(n) : \mathbb{N} \rightarrow \mathbb{N}$ be the function mapping the input size to the maximum message size. The restriction of $\text{QMIP}_{ne}(k)$ to messages of length at most $O(f(n))$ is denoted by $\text{QMIP}_{ne}(k)_{f(n)}$. For arbitrary completeness and soundness c and s , we denote $\text{QMIP}_{ne}(k, c, s)$, the version of $\text{QMIP}_{ne}(k)$ with these parameters. Putting it all together, for each tuple $(\hat{s}, f(n), c, s)$, we have the class $\hat{s}\text{-QMIP}_{ne}(k, c, s)_{f(n)}$. Note that without loss of generality, any parallel interaction such as $[qq]_1 \parallel \cdots \parallel [qq]_k$ can be made sequential as $[qq]_1 \rightarrow \cdots \rightarrow [qq]_k$ without reducing the expressive power of the QMIP_{ne} variation. The converse is not necessarily true.

When working with variations of QMIP_{ne} , it is necessary to have in mind the classical result about MIP that shows that this kind of proof system already reaches all its power when very few resources are used. To be more precise, a two player game, in which the referee queries have polynomial size and there are only a constant number of possible answers, is enough to show the inclusion $\text{NEXP} \subseteq \text{MIP}(= \text{NEXP})$. Furthermore, in this game, the referee makes only two queries in parallel. In terms of our syntax, this result can be stated as $\text{NEXP} \subseteq [cc]_1 \parallel [cc]_2 - \text{MIP}(2)$. For this reason, when extending $\text{QMA}(k)$ with interaction it is crucial not to make the proof system exceedingly powerful, in which case it will trivially contain NEXP .

7.4 Alternative Characterization of $\text{QMA}(k)$

Beige et al. showed that the class QMA remains the same even if the verifier is allowed to send a logarithmic size query to the prover who answers with a regular polynomial size state. This extra flexibility might be useful when designing new QMA protocols. Formally, their result can be stated as the following theorem.

Theorem 7.4.1 (From [15]). *Let $a, b : \mathbb{N} \rightarrow [0, 1]$ be polynomial-time computable functions such that $a(n) - b(n) \geq \frac{1}{p(n)}$ for some polynomial p . Then $\text{QIP}([\log, poly], a, b) = \text{QMA}$.*

The class $\text{QIP}([q, r], a, b)$ stands for a variation of QIP where the verifier queries the prover using q -qubits and gets a reply from the prover with r -qubits. The completeness and soundness are given by a and b , respectively.

We show how to extend their result to $\text{QMA}(k)$, in lieu of QMA . Instead of being able to issue at most one query, the verifier can query a constant number of provers T provided that the total query length remains logarithmic. This new setting provides two interesting characteristics that were not present in the previous case. First, the verifier

can condition a latter query on the result of previous ones. Second, it may have at its disposal polynomially many provers even though it can query at most a constant number of them.

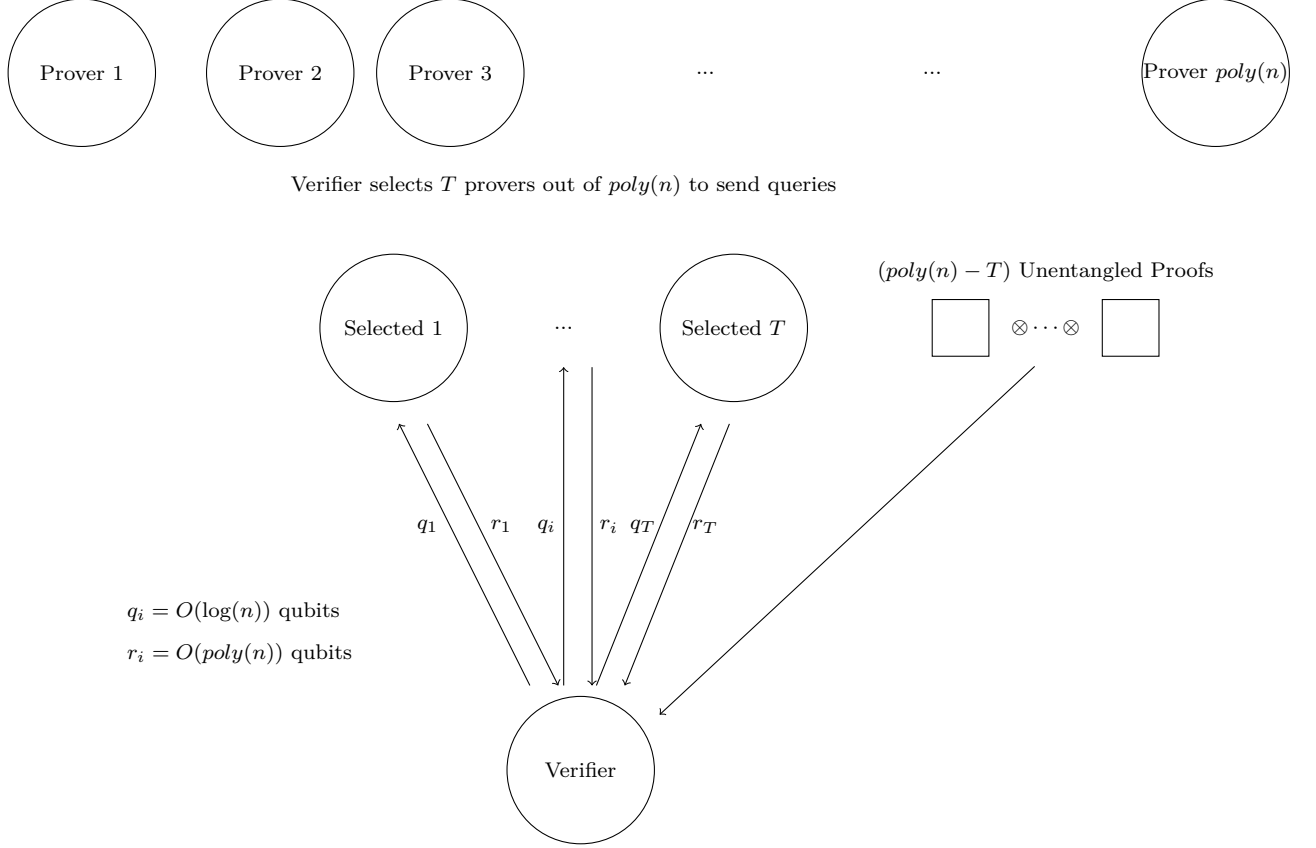


Figure 7.1: Beige et al. short message QMA protocol extended to QMA(k).

Upon receiving a query state $\sigma \in \mathcal{Q}$ on q -qubits, the prover applies a quantum channel $\Phi : \mathcal{Q} \rightarrow \mathcal{R}$. Note that the state σ may be entangled with another system. As showed in 3.7, a quantum channel Φ of this form corresponds to a state $\psi_\Phi \in \mathcal{Q} \otimes \mathcal{R}$ known as its Choi-Jamiolkowski representation. From this state ψ_Φ and σ , it is possible to simulate the application of Φ to σ by post-selection with a success probability of at least $\frac{1}{4^q}$, as shown in 4.4. As long as q has logarithmic size with respect to the input length n , the interaction with a trustful prover could be replaced by this Choi-Jamiolkowski state which implies that the channel can be successfully simulated with probability at least $\frac{1}{\text{poly}}$. However, in our setting the prover is adversarial, which prohibits the assumptions that the received state is a Choi-Jamiolkowski state. Using versions of the quantum de Finetti Theorem, and quantum state tomography, the verifier is able to ensure that the post-

selection process will have an inverse polynomial probability of success. The following procedure combines both the state verification and the channel simulation.

- 1 Input: state σ , $N + m$ registers $(R_1, Q_1), \dots, (R_{N+m}, Q_{N+m})$ with N and m polynomials in the input length n , to be shown below.
- 2 Apply a random permutation $\pi \in S_{N+m}$ to $(R_1, Q_1), \dots, (R_{N+m}, Q_{N+m})$, and discard all except the first $N + 1$ pairs.
- 3 Perform quantum state tomography on the registers (Q_2, \dots, Q_{N+1}) rejecting if $\|H - \frac{1}{2^q} \mathbb{I}\|_1 > \frac{\delta}{2}$, where H is the approximate state and δ is specified below.
- 4 Simulate a quantum channel with query σ using the pair (R_1, Q_1) . Reject if the post-selection fails. Otherwise, return the channel output.

Algorithm 8: Quantum Channel Simulation (adapted from [15]).

Without loss of generality, we assume that all channels act on q -qubits as input. We set the parameters so that the tomography error is $\frac{\delta}{2}$, and the quantum de Finetti error is at most ϵ . This requires the following values

$$\epsilon = \frac{1}{pT4^{Tq+2}}, \delta = \frac{\epsilon^2}{4},$$

$$N = \frac{2^{10q}}{(\delta/2)^3}, m = \frac{2N4^q}{\epsilon}.$$

We calculate the quantum channel simulation success probability in the case the input $(R_1, Q_1), \dots, (R_{N+m}, Q_{N+m})$ is given by a trustful prover. In this case, these registers actually contain the same Choi-Jamiolkowski state. Thus their reduced state on Q_i is the totally mixed state. The quantum state tomography gives an approximation H within $\frac{\delta}{2}$ in the trace distance from $\frac{1}{2^q} \mathbb{I}$, with probability at least $1 - \frac{\delta}{2}$. Conditioned on not being rejected after tomography, the post-selection probability is $\frac{1}{4^q}$. For one channel simulation, the total success probability is

$$(1 - \frac{\delta}{2}) \frac{1}{4^q}.$$

Now, we analyze the case in which $(R_1, Q_1) \dots (R_{N+m}, Q_{N+m})$ is given by an adversarial prover. First, suppose that after the application of the random permutation on line 2, we have the product state ξ^{N+1} . Using this state in the quantum tomography, there are two cases to consider. The first one is when $\|\xi - \frac{1}{2^q} \mathbb{I}\|_1 \leq \frac{\delta}{2}$. In this case, we want to show that the state on (R_1, Q_1) , which we denote by ϕ , is close to a state ρ where $\text{Tr}_{R_1}(\rho) = \frac{1}{2^q} \mathbb{I}$. To this end, we use both Fuchs-van de Graaf inequalities

$$1 - F(\rho, \sigma) \leq \|\rho - \sigma\|_1 \leq \sqrt{1 - F(\rho, \sigma)^2}.$$

Let $|\phi\rangle$ be a purification of ϕ on (R_1, Q_1, E) , where E is a reference system with the same number of qubits as (R_1, Q_1) . Using the first inequality, we have

$$F(\text{Tr}_{R_1}(\rho), \text{Tr}_{R_1 E}(|\phi\rangle\langle\phi|)) \geq 1 - \frac{\delta}{2}.$$

We use the state $|\psi\rangle$ to denote a purification of ρ on (R_1, Q_1, E) . Using an alternative definition of fidelity in terms of purifications, we have

$$F(\text{Tr}_{R_1}(\rho), \text{Tr}_{R_1 E}(|\phi\rangle\langle\phi|)) = \max_{|\psi\rangle} |\langle\psi|\phi\rangle| \geq 1 - \frac{\delta}{2}.$$

Denote by $|\psi^*\rangle$ the optimum purification for ρ in this maximization. Note that the fidelity of the reduced state on (R_1, Q_1) of these purifications is

$$\begin{aligned} F(\text{Tr}_E(|\psi^*\rangle\langle\psi^*|), \text{Tr}_E(|\phi\rangle\langle\phi|)) &= F(|\psi^*\rangle\langle\psi^*|, |\phi\rangle\langle\phi|) \\ &= |\langle\psi^*|\phi\rangle| \geq 1 - \frac{\delta}{2}. \end{aligned}$$

Now, using the second Fuchs-van de Graaf inequality, we can calculate

$$\begin{aligned} \|\text{Tr}_E(|\psi^*\rangle\langle\psi^*|) - \text{Tr}_E(|\phi\rangle\langle\phi|)\|_1 &\leq \sqrt{1 - F(\text{Tr}_E(|\psi^*\rangle\langle\psi^*|), \text{Tr}_E(|\phi\rangle\langle\phi|))} \\ &\leq \sqrt{1 - (1 - \delta)} = \frac{\epsilon}{2} \leq \epsilon. \end{aligned}$$

Since $\rho = \text{Tr}_E(|\psi^*\rangle\langle\psi^*|)$ and $\phi = \text{Tr}_E(|\phi\rangle\langle\phi|)$, this result shows that the actual state on (R_1, Q_1) is within an ϵ trace distance from ρ . In this case, assume for simplicity that there is no rejections after the tomography. On ρ , the post-selection succeeds with probability $\frac{1}{4^q}$. Since ϕ is not equal to ρ , we attribute their ϵ trace distance to the total acceptance probability that the dishonesty prover can take advantage of.

The other case occurs when $\|\xi - \frac{1}{2^q}\mathbb{I}\|_1 > \frac{\delta}{2}$. From 4.2.1, the quantum state tomography has a probability at most $\frac{\delta}{2}$ of returning a description $\|H - \xi\|_1 < \frac{\delta}{2}$. Assume that in all ϵ ($> \frac{\delta}{2}$) fraction of the cases, they lead to verifier acceptance. In both cases, the prover has an advantage of ϵ over a trustful prover.

The quantum de Finetti Theorem guarantees only a state close to a convex combination of product states of the form

$$\sum_j p_j \xi_j^{N+1}.$$

From our parameters, the actual state on the $N + 1$ pairs is within ϵ trace distance from a state of the above form. Therefore, the dishonesty prover's advantage is bounded by 2ϵ .

The convex combination of product states from the de Finetti Theorem can be thought of as a probabilistic distribution on channels. By convexity, there will be a choice of

channels that maximizes the verifier's acceptance probability. Therefore, the prover has no advantage in giving the verifier a convex combination.

Without loss of generality, we also assume that the verifier issues $T \in O(1)$ queries. In the QMA(k) protocol, each time it would issue a query the previous channel simulation procedure is applied. The complete verifier action is described next.

- 1 For each of the T times the verifier would send a query σ to a prover i , simulate a quantum channel on this state using procedure 8.
- 2 Accept if and only if all channel simulations succeeded and the original protocol would have accepted.

Algorithm 9: QMA(k) protocol for QMIP_{ne}(k, a, b).

Let L be a language in QMIP_{ne}(k, a, b) and $x \in \{0, 1\}^n$. If $x \in L$, the analysis of the trustful prover applies. The acceptance probability after T queries is at least

$$[(1 - \frac{\delta}{2}) \frac{1}{4^q}]^T a \geq (1 - T \frac{\delta}{2}) (\frac{1}{4^q})^T a = (\frac{1}{4^q})^T a - T \frac{\delta}{2} (\frac{1}{4^q})^T a,$$

where the first inequality follows from the Bernoulli inequality. If $x \notin L$, the dishonesty prover advantage accumulates to $2T\epsilon$ after T queries. The acceptance probability is at most

$$(\frac{1}{4^q})^T b + 2T\epsilon.$$

Calculating the completeness soundness gap as

$$(\frac{1}{4^q})^T \frac{1}{p} - (\frac{1}{4^q})^T T \frac{\delta}{2} a - 2T\epsilon,$$

and replacing the parameter ϵ by its value, we have

$$\begin{aligned} & (\frac{1}{4^q})^T \frac{1}{p} - (\frac{1}{4^q})^T \frac{1}{2p^2 T 4^{2Tq+4}} a - \frac{1}{4p 4^{Tq}} \\ & \geq (\frac{1}{4^q})^T \frac{1}{p} - (\frac{1}{4^q})^T \frac{1}{4p} - (\frac{1}{4^q})^T \frac{1}{4p} \\ & = (\frac{1}{4^q})^T \frac{1}{2p} \end{aligned}$$

which is still inversely polynomial for a constant T .

The result proved above can be rephrased in our QMIP_{ne} syntax as the following theorem.

Theorem 7.4.2. QMA = $[q]_{T+1} \rightarrow \cdots \rightarrow [q]_{poly} \rightarrow ([qq]_1 \rightarrow \cdots \rightarrow [qq]_T)$ -QMIP($poly, c, s$) where n is the input size, $c - s \geq \frac{1}{p(n)}$ for some polynomial p , and $T \in O(1)$.

Note 7.4.3. *If the query size had polynomial length, this construction would imply that $\text{NEXP} = \text{QMA}(k)$. However, in the way it was done it does not seem evident how to increase the query size. The quantum tomography and the de Finetti theorem would require exponentially many copies.*

7.5 One Classical Query and The $[cq]_1 \parallel [q]_2 - \text{QMIP}_{ne}(2)$ Class

From the previous result, a logarithmic size query from the verifier to the prover does not increase $\text{QMA}(k)$. A natural question is: what happens when one classical polynomial size query is allowed? This leads to the extension $[cq]_1 \parallel [q]_2 - \text{QMIP}_{ne}(2)$. By virtue of the interaction $[cq]_1$, it is possible to recognize the Graph-Non-Isomorphism (GNI) language, which is not known to be in $\text{QMA}(2)$. This extension also contains any language in SZK, AM, and QAM. For this reason, this extra polynomial classical query might make $\text{QMA}(k)$ larger. Nevertheless, we don't know any more interesting lower bounds.

7.6 Quantum Random Query and the $[eq]_1 \parallel [q]_2 - \text{QMIP}_{ne}(2)$ Class

This is a variation of $[cq]_1 \parallel [q]_2 - \text{QMIP}_{ne}(2)$ in which the first interaction consists in the verifier sending EPR halves to the prover. This class also contains QAM. Note that interactions of the form $[eq]_i$ can be replaced by $[ec]_i$, using teleportation. Thus this class is equal to $[ec]_1 \parallel [q]_2 - \text{QMIP}_{ne}(2)$.

7.7 The Containment $\text{NP} \subseteq [q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}(2)_{\log(n)}$

We show that as long as the messages are quantum, by receiving one message from a prover and then having one round of interaction with another unentangled prover, it is enough to verify languages in NP, or NEXP, depending on the message size. Note that the classical analogue of $[q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}(2)$ is $[c]_1 \rightarrow [cc]_2 - \text{MIP}(2)$, which is equal to IP[3]. This fact provides a good indication that quantum information is more powerful. We provide a simple protocol showing this result, but we observe that Ran Raz has an improved protocol in which the second prover is classical, and whose analysis is more involving.

As in [3], given an instance φ of 3SAT, we make two transformations. Firstly, we amplify the soundness gap to a constant, using Dinur's method [29]. Then, we use Pa-

padimitriou and Yannakakis method to make every variable appear in exactly 29 clauses without harming the soundness gap [3]. The first method has a polylogarithmic blowup in the instance size, while the second has only a constant one. Thanks to these methods, we can then use a protocol similar to those used for the outer verifier in PCP Theorems [48].

After these transformations, let φ' denote the new instance of 3SAT. As we deal with EPR pairs, we have to overcome a simple technicality that consists in making the number of clauses a power of 2. If this number is not a power of 2, then we add dummy clauses of the form $C^i = (x_1^i, x_2^i, x_3^i)$, in which x_1^i , x_2^i and x_3^i are dummy variables that only appear in C^i until the number of clauses becomes a power of two. We denote by D the set of dummy clauses. Using this process, the number of clauses at most doubles, so the gap is at least half the original gap. Note that φ' with dummy clauses is satisfiable if and only if φ is. Let M and N be the final number of clauses and the final number of variables, respectively.

The verifier expects from the first prover a superposition in the form

$$|\psi\rangle = \sum_{i \in [M]} \frac{1}{\sqrt{M}} |i\rangle |i\rangle |(l_1^i, l_2^i, l_3^i)\rangle, \quad (7.1)$$

where for each clause i , $C^i = |(\hat{l}_1^i, \hat{l}_2^i, \hat{l}_3^i)\rangle$ is the corresponding boolean assignment provided to its literals.

The verifier performs two tests. The first one ensures that the probability distribution of measuring the index part of the received state in the computational basis is close to uniform. It does so by preparing M EPR pairs, sending M halves to the second prover which adds an ancilla $|000\rangle$, applies the unitary transformation $|i\rangle|000\rangle \rightarrow |i\rangle|(l_1^i, l_2^i, l_3^i)\rangle$ and send the result back to the verifier. Let ϕ denote the density matrix of the system composed of the EPR pairs that remained in possession of the verifier with the state received from the second prover. Next, the verifier performs a swap test with the states $|\psi\rangle$ and ϕ . This test ensures that ϕ is close to a pure state $|\phi\rangle$. As $|\phi\rangle$ is close to a purification of the EPR halves that remained in the possession of the verifier, it is close to a state of the form $\sum_{i \in [M]} \frac{1}{\sqrt{M}} |i\rangle^A U^{BC} |i\rangle^B |0\rangle^C$. Moreover, the swap test also ensures that the probability distribution of measuring $|\psi\rangle$ and $|\phi\rangle$ in the computational basis is close.

Now, the second test can safely assume that the state $|\psi\rangle$ will yield an unsatisfiable clause with constant probability in case there is no satisfying assignment for the 3SAT instance. Suppose we measured one of them in which the prover must have lied about the assignment of at least one of its literals, otherwise the verifier would have readily rejected. Then the verifier chooses uniformly at random a variable from this clause and query the second prover about its value. As each variable appears in at most 29 clauses, there is

a constant probability of at least $\frac{1}{3} \frac{1}{29}$ of detecting that the prover is cheating given an unsatisfiable clause. Overall, there is a constant probability of uncovering a malicious prover. Instead of relying on the fact that each variable appears in a constant number of clauses, we could have done the analysis in terms of the second player's strategy, part of which constitutes a fixed assignment.

Theorem 7.7.1. *3SAT is contained in $[q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}(2, 1, 1 - \epsilon)_{\log n}$ for some constant ϵ .*

The verifier protocol just described is depicted in 10.

Input: Receive state $|\psi\rangle$ from the first prover.
 With equal probability perform one of the following actions:
 1) (Check Randomness)
 1.1) Prepare M EPR pairs as $\sum_{i \in [M]} \frac{1}{\sqrt{M}} |i\rangle^A |i\rangle^B$;
 1.2) Send system B to the second prover; denote by ϕ the state of system A and the state received back from the second prover ;
 1.3) Perform a swap test with $|\psi\rangle$ and $|\phi\rangle$, accepting iff it accepts ;
 2) (Perform the NP protocol)
 2.1) Measure the proof $|\psi\rangle$ to get a clause i with (l_1^i, l_2^i, l_3^i) ;
 2.2) If $(l_1^i, l_2^i, l_3^i) = (0, 0, 0)$, then reject ;
 2.3) Choose uniformly at random a variable j from clause i ;
 2.4) Ask the second prover the value of j ;
 2.5) Accept iff the value received for j is the same as in clause i ;

Algorithm 10: $[q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}(2)_{\log n}$ verifier for NP

In case the input belongs to the language, the honest prover is given by the actions in 11.

1) (Check Randomness)
 1.1) Apply the unitary U that transforms $|i\rangle|000\rangle \rightarrow |i\rangle|(l_1^i, l_2^i, l_3^i)\rangle$;
 1.2) Send the resulting state to the verifier;
 2) (Perform the NP protocol) Give the correct value of variable j ;

Algorithm 11: $[q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}(2)_{\log n}$ honest prover for NP

7.7.1 Completeness

The first prover sends a state $|\psi\rangle$ as in 7.1. The randomness check never rejects if the second prover applies the appropriate unitary. Furthermore, in the NP protocol, as all clauses are satisfiable, the second prover will always be able to answer correctly the value of the queried variable. Consequently, the quantum protocol has perfect completeness.

7.7.2 Soundness

As explained, if the probability of measuring an unsatisfiable clause is constant in a no instance, then the NP protocol rejects with constant probability. Suffices us to show that if this is not the case, the randomness check fails with constant probability.

To simplify the analysis, assume that with probability $\frac{1}{2}$ the randomness check enforces that ϕ is close to pure, and with probability $\frac{1}{2}$ it enforces that the distributions of measuring the two states $|\psi\rangle$ and ϕ in the computational basis are close.

In the first case, for any constant $\epsilon < 1$, if ϕ is ϵ -far in the trace distance from a pure state, the swap test fails with constant probability. Assuming ϕ is ϵ -close to $|\phi\rangle$, ϕ is ϵ -close to $|\xi\rangle = \sum_{i \in [M]} \frac{1}{\sqrt{M}} |i\rangle^A U^{BC} |i\rangle^B |0\rangle^C$, for some unitary U^{BC} , as $|\xi\rangle$ is a purification of the EPR halves that remained with the verifier.

Let δ be the fraction of the non satisfiable clauses. Suppose ϕ is $\frac{\delta}{100}$ -close to $|\xi\rangle$. If the probability of measuring such a clause using $|\psi\rangle$ in the computational basis is less than $\frac{\delta}{2}$, the distance of $|\xi\rangle$ and $|\phi\rangle$ is at least $\frac{\delta}{2} - \frac{\delta}{100}$. Also, the probability of rejecting in the swap test is a constant. As the verifier has perfect completeness, the protocol has a constant completeness soundness gap.

7.7.3 Scaling Up

There is a scaled-up version of the PCP Theorem for NEXP in which the verifier has an exponentially long proof, uses polynomially many random bits, and queries only a constant number of positions in this proof [70]. For this reason, the protocol we just devised for NP can be extended to NEXP as well, but now with polynomial size messages. This results in the following theorem.

Theorem 7.7.2.

$$\text{NEXP} \subseteq [q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}(2).$$

Recall that to show the containment $\text{NEXP} \subseteq \text{MIP}$, the verifier needs to query both provers. In the quantum case, it is enough to have a proof from the first prover and query just the second. There is an even more restricted class than $[q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}(2)$ that already contains NEXP which has a more involved proof, as discussed in 7.10.

7.8 The Containment:

$$\text{PSPACE} \subseteq [q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}(2, 1 - 2^{-poly}, 2^{-poly})$$

In classical Interactive Proof systems ($\text{IP}[k]$), we know that $\text{IP}[3] \neq \text{PSPACE}$, unless the Polynomial Hierarchy collapses [62]. Surprisingly, in the quantum case, $\text{QIP}[3] =$

PSPACE as shown by Watrous et al. in [58]. In this section, we show the following theorem.

Theorem 7.8.1. $\text{PSPACE} \subseteq [q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}(2, 1 - 2^{-poly}, 2^{-poly})$.

This result is superseded by our previous result, and certainly by Ran Raz 7.10 as well. However, the proof of our results revolves around the utilization of EPR pairs which is, in some sense, a quantum analogue of randomness, and the underlying ideas can be useful elsewhere. In this proof, we use the relation of Goldwasser-Sipser that $\text{IP}[k] = \text{AM}[k + 2]$.

Firstly, we show that $\text{AM}[2m] \subseteq [q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}(2, 1 - 2^{-poly}, 1 - \frac{1}{poly})$ where the completeness soundness gap is an inverse polynomial. Then, a parallelization lemma makes the error exponentially small.

In an interactive proof systems, at each round the prover must answer a verifier's query without knowing the future ones. This property seems essential to ensure a controlled soundness for this kind of system. In the specific case of $\text{AM}[k]$, the system relies on the fact that at each round the query is a random string drawn from a uniform distribution not letting the prover anticipate the verifier's actions. In the quantum setting, states of polynomially many qubits lie in exponentially large Hilbert spaces, allowing them to encode a superposition of any $\text{AM}[k]$ prover's strategy. For $\text{AM}[2m]$, the next state encodes a valid prover strategy

$$|\psi\rangle = \sum_{r_1 \in \{0,1\}^{s_1}} \frac{1}{\sqrt{2^{s_1}}} |r_1\rangle |r_1\rangle |m_1\rangle \sum_{r_2 \in \{0,1\}^{s_2}} \frac{1}{\sqrt{2^{s_2}}} |r_2\rangle |r_2\rangle |m_{12}\rangle \cdots \sum_{r_m \in \{0,1\}^{s_m}} \frac{1}{\sqrt{2^{s_m}}} |r_m\rangle |r_m\rangle |m_{1\dots m}\rangle, \quad (7.2)$$

where s_i is the number of random bits in round i . The verifier simulates each round i by measuring the registers $\sum_{r_i \in \{0,1\}^{s_i}} \frac{1}{\sqrt{2^{s_i}}} |r_i\rangle |r_i\rangle |m_{1\dots i}\rangle$ in the computational basis. It is clear that the underlying probability distribution is uniform as in the original protocol. Moreover, the prover can use the history of the previous outcomes and the current random string to determine $|m_{1\dots i}\rangle$, as it is allowed classically.

Unfortunately, the norm of these amplitudes are exponentially small, making it difficult for a polynomial time verifier to attest if the prover has actually sent a state in the form of 7.2. For this reason, we make use of another unentangled prover. We send to this new prover in round i , the history $|x_i\rangle = |r_1\rangle \dots |r_{i-1}\rangle$ and s_i EPR halves, and it returns the state we should expect after simulating $i - 1$ rounds and having seen history $|x_i\rangle$.

Before we proceed, we assume, without loss of generality, that the number of messages exchanged in the $\text{AM}[k]$ is even. Moreover, by sequentially repeating the classical protocol a polynomial number of times and applying Chernoff's bound, we can assume that it has completeness and soundness errors of at most 2^{-poly_e} .

The verifier protocol is given in 12.

Input: Receive state $|\psi\rangle$ from first prover.

With equal probability perform one of the following actions:

1) (Check Randomness)

1.1) Choose i from 1 to m and do:

1.2) Simulate $i - 1$ rounds by measuring the first $i - 1$ registers, corresponding to $|r_1\rangle|r_1\rangle|m_1\rangle \dots |r_{i-1}\rangle|r_{i-1}\rangle|m_{i-1}\rangle$ (denote the resulting state by $|\psi'_i\rangle$) ;

1.3) Prepare s_i EPR pairs as $\sum_{r_1 \in \{0,1\}^{s_i}} \frac{1}{\sqrt{2^{s_i}}} |r_i\rangle^{A_i} |r_i\rangle^{B_i}$;

1.4) Set $|x_i\rangle = |r_1\rangle \dots |r_{i-1}\rangle$ to be the classical state corresponding to the measured random bits ;

1.5) Send system B_i and history $|x_i\rangle$ to the second prover ;

1.6) Let ϕ be the state composed of system A and the state received back from the second prover ;

1.7) Perform the swap test with $|\psi'\rangle$ and ϕ accepting iff it accepts;

2) (Perform the AM[2m] protocol)

Use $|\psi\rangle$ to perform the AM[2m] protocol. Simulate sequentially the m rounds, collecting the random bits and the messages. Then, apply the classical verification procedure accepting iff it accepts.

Algorithm 12: $[q]_1 \rightarrow [qq]_2$ – QMIP_{ne}(2) verifier for AM[2m].

7.8.1 Completeness

It is straightforward to check that the completeness of the protocol is at least $1 - 2^{-poly_e}$. Test 1 never fails and test 2 accepts with probability at least $1 - 2^{-poly_e}$.

7.8.2 Soundness

Note that for the classical AM[2m] verifier to work correctly, the only requirement is that at each round the associated random string is drawn from a distribution close to uniform with high enough probability. This ensures that in all m rounds there is a high probability of selecting random bits as if they were drawn from the uniform distribution, not disturbing much the accepting and rejecting probabilities.

We define $p(r_1, \dots, r_m)$ to be the probability distribution of measuring r_1, \dots, r_m as the random bits. Let $p(r_1, \dots, r_{i-1})$ be the marginal probability distribution given by

$$\sum_{r_i \in \{0,1\}^{s_i}} \dots \sum_{r_m \in \{0,1\}^{s_m}} p(r_1, \dots, r_{i-1}, r_i, \dots, r_m).$$

Let $D(p(r_1, \dots, r_{i-1}, _), U_{2^{s_i}})$ be the statistical distance of the marginal $p(r_1, \dots, r_{i-1}, r_i)$ when fixing all except the last variable, with the uniform probability distribution $U_{2^{s_i}}$ having 2^{s_i} outcomes. We define Y_i to be the random variable given by $D(p(r_1, \dots, r_{i-1}, _), U_{2^{s_i}})$ occurring with probability $p(r_1, \dots, r_{i-1})$.

Lemma 7.8.2. *For every polynomial $poly_1 : \mathbb{N} \rightarrow \mathbb{N}$, if there is an i in $[m]$ such that $\mathbf{E}[Y_i] > \frac{1}{poly_1}$, then test 1 fails with probability at least $\frac{1}{poly'_1}$, for some polynomial $poly'_1$.*

Proof. The utility of the swap test in the randomness check is twofold. Firstly, it checks that the state sent by the second prover is close to pure. Secondly, it compares the closeness of the distributions associated with measuring r_i using $|\psi'_i\rangle$ and the uniform $U_{2^{s_i}}$.

The reduced state of $\sum_{r_i \in \{0,1\}^{s_i}} |r_i\rangle^A |r_i\rangle^B$ on system A is a totally mixed state, and the state received from the second prover, after applying some channel to system B , gives a state that is close to a purification of A . This state is denoted by ϕ and it is $\frac{1}{poly_p}$ -close to $\frac{1}{\sqrt{2^{s_i}}} |r_i\rangle^A U^{BC} |r_i\rangle^B |0\rangle^C$, for some unitary U acting on system BC , or otherwise the swap test would fail with probability at least $\frac{1}{poly'_p}$. This state can be used as an approximation of $U_{2^{s_i}}$ to test the uniformity of $p(r_1, \dots, r_{i-1}, _)$.

Suppose there is an $i \in [m]$ satisfying $\mathbf{E}[Y_i] > \frac{1}{poly_1}$. Lemma 7.8.2 states that there is an $\epsilon = \frac{1}{poly'_1} < \frac{1}{poly_1}$ probability of Y_i being greater than $\delta = \frac{1}{poly_1} - \frac{1}{poly'_1}$. Round i is selected with probability $\frac{1}{m}$, in which case with probability ϵ the statistical distance of $p(r_1, \dots, r_{i-1}, _)$ from uniform, represented by Y_i , is at least δ . As we are using only an approximation to $U_{2^{s_i}}$, the statistical distance is $\delta' \geq \delta - \frac{1}{poly_p} > 0$. Then, the swap test fails with probability at least $\frac{\epsilon}{m} \frac{1}{2} \delta'^2 = \frac{1}{poly'_d}$. Consequently, test 1 fails with probability at least $\frac{1}{poly'_1} = \min\{\frac{1}{poly'_p}, \frac{1}{poly'_d}\}$ \square

Lemma 7.8.3. *Suppose the input does not belong to the language L of $\text{AM}[2m]$. If for all $i \in [m]$, it holds that $\mathbf{E}[Y_i] < \frac{1}{poly_1}$, then test 2 rejects it with probability at least $1 - \frac{1}{poly_2}$.*

Proof. Using to the randomness check, for any polynomial $poly_1$ it holds that $\mathbf{E}[Y_i] < \frac{1}{poly_1}$. Using the Markov inequality ($P[X \geq \lambda] \leq \frac{\mathbf{E}[X]}{\lambda}$), it is possible to bound the probability of Y_i being greater than an inverse polynomial as $P[Y_i \geq \frac{1}{poly}] \leq \mathbf{E}[Y_i] poly = \frac{1}{poly'_1}$. This means that each time the verifier measures the random bits in $|\psi'_i\rangle$ there is a probability at least $1 - \frac{1}{poly'_1}$ that the statistical distance from uniform be smaller than $\frac{1}{poly}$, implying that, with total probability $[(1 - \frac{1}{poly})(1 - \frac{1}{poly'_1})]^m > (1 - \max\{\frac{1}{poly}, \frac{1}{poly'_1}\})^{2m}$, the random bits for the classical verifier will be drawn as if they were from the uniform distribution. The Bernoulli inequality $((1+x)^n \geq 1+nx)$ shows that, by carefully choosing $\frac{1}{poly_1}$ and $\frac{1}{poly}$, we have a probability of $1 - \frac{1}{poly'''}$ of not noticing any difference from uniform in all rounds. Finally, the $\text{AM}[2m]$ verifier rejects with probability $1 - \frac{1}{poly_2} \geq (1 - \frac{1}{poly'''})(1 - 2^{-poly_e})$. \square

Combining the two previous lemmas we see that soundness is an inverse polynomial.

7.9 Parallelization

In order to amplify the completeness soundness gap of the protocol, we run it polynomially many times in parallel. In this case, the randomness check still applies, as Lemma 7.8.2 remains unchanged. Consequently, it suffices to collect the outcomes of parallel executions and use a Chernoff bound to make the error becomes exponentially small.

7.10 Quantum Proof Followed by a Classical Round: the class $[q]_1 \rightarrow [cc]_2 - \text{QMIP}_{ne}(2)$

Ran Raz showed that the class $[q]_1 \rightarrow [cc]_2 - \text{QMIP}_{ne}(2)$ contains NEXP [83]. Measuring one quantum proof and then having a single round of interaction with a classical prover is enough to unleash all the power of QMIP_{ne} , since the latter is contained in NEXP. Clearly, Raz's result supersedes ours for the class $[q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}$. Nonetheless, we chose to include our proofs because they are simpler and may give some ideas that might be helpful in other contexts ¹.

Theorem 7.10.1 (From Raz [83]).

$$\text{NEXP} \subseteq [q]_1 \rightarrow [cc]_2 - \text{QMIP}_{ne}(2).$$

¹The author also discovered the proofs $[q]_1 \rightarrow [qq]_2 - \text{QMIP}_{ne}$ before encountering Raz's paper. Despite being a very interesting contribution, this paper has been largely ignored by subsequent related works.

Chapter 8

Classical Probabilistically Checkable Proofs and Multi-Prover Quantum Merlin-Arthur

8.1 Introduction

Witness verification protocols have a relevant role in Computational Complexity since they capture several problems of practical interest [41]. Languages whose membership can not be even certified in polynomial time are beyond polynomial time decidability. The famous P vs. NP question asks the converse: can all efficient certifiable languages be efficiently decidable? With the advent of quantum computing [36], witness verification protocols have been extended and adapted to this new quantum model. Unique quantum mechanical properties lead to models that bare no resemblance to classical ones. Classically, having multiple witnesses does not change the computational power. However, in the quantum case, can the lack of entanglement make the system more powerful?

This question was firstly studied by Kobayashi et al., who introduced the class $\text{QMA}(k)$ [65]. It captures the notion of k unentangled quantum provers. An entanglement correlation is a unique quantum mechanical property that can not be described classically, as attested in practice by violations of the Bell inequalities [86]. These super classical correlations can be an important resource in certain quantum information processing tasks such as superdense coding and teleportation [98]. Nonetheless, in the context of Multi-prover proof systems entanglement shared among the provers might harm their soundness. Exploring the hypothesis that the proofs are not entangled may lead to distinctive features of theses systems. Understanding these features, creating methods to detect entanglement, or even to break it [3], are important research directions underlying the study of the complexity class $\text{QMA}(k)$.

Blier and Tapp conceived a verifier protocol in $\text{QMA}(2)_{\log(n)}$ for the vertex three-coloring problem on graphs (*3COL*) [17]. Since this is a NP-complete problem, it follows that $\text{NP} \subseteq \text{QMA}(2)_{\log(n)}$. This means that it suffices to have two unentangled quantum proofs of logarithm size to verify if a language is in NP. This type of protocol is an indication that unentanglement might increase the computational power of a proof system.

In this work, we extend Blier and Tapp's protocol to create a generic $\text{QMA}(k)$ verifier capable of simulating classical PCP verifiers. With this generalization, several classical PCP results can benefit from an exponential proof size reduction in the quantum setting. On the other hand, this dramatic reduction comes at the cost of reducing the completeness-soundness gap. Furthermore, we show that it is possible to recover two known results in a simplified way. Firstly, a $\text{QMA}(2)_{\log(n)}$ verifier for *3SAT* with completeness-soundness gap of $\Omega(\frac{1}{n^{2+\varepsilon}})$ for any $\varepsilon > 0$, which lies between the Beigi [14] and the Le Gall et al. [40] results. Secondly, we also have a proof of $\text{QMA}(2) = \text{NEXP}$ in the context of a small gap [23].

The organization of the chapter is as follows. In Section 8.2, related results about Quantum Merlin-Arthur Multi-provers are discussed. In Section 8.3, some preliminary definitions and theorems are stated. Our main result, the verifier protocol that simulates a generic PCP classical verifier, is presented in Section 8.4, along with some implications for two concrete classical PCP theorems. Finally, in Section 8.5 we discuss some open questions.

8.2 Related Results

The discovery of NP-complete verifier protocols in the context of unentangled Multi-prover Quantum Merlin-Arthur provers, with logarithmic proof size, attracted much attention to the complexity class $\text{QMA}(k)$ and its variations. These variations may concern the number of provers, the proof size, and the type of measurements used by the verifier. Even though a large number of results are now known, tight bounds for this class remain major open questions. In this section, we briefly survey some of these results and we highlight some connections with our work.

What makes $\text{QMA}(k)$ surprising is that if protocols such as the one of Blier and Tapp [17] were true for the classical Multi-prover case, it would imply that $\text{P} = \text{NP}$. A brute force algorithm would just need to enumerate all possible proofs and check them. Given that the proof length is logarithmic, the total number of proofs is polynomial in the input size. As a result, this algorithm would require only polynomial time. However, in the quantum case we do not know how to search separable (not entangled) states efficiently [20]. In fact, the problem of deciding if a classical description of a density operator is close to a separable one was shown to be NP-hard, when the distance depends on an inverse

polynomial in the dimension [42].

Aaronson et al. showed that with $O(\sqrt{n} \log(n))$ proofs of logarithm size it is possible to verify a $3SAT$ instance with a constant completeness-soundness gap [3]. The original protocol of Blier and Tapp (BT) used only two proofs and achieved a gap of $\Omega(\frac{1}{n^6})$ for $3COL$ [17]. Subsequent works have improved this gap. Beigi devised a protocol for $3SAT$ with a gap of $\Omega(\frac{1}{n^{3+\epsilon}})$ [14]. Afterwards, Chiesa et al., using the same original ideas of BT, improved the previous analysis for $3COL$ to a gap of $\Omega(\frac{1}{n^2})$ [23]. Finally, Le Gall et al. using the BT protocol improved the gap to $\Omega(\frac{1}{n^{\text{polylog}(n)}})$ for $3SAT$, but now combined with a gapped Constraint Satisfaction instance from Dinur's PCP Theorem [29] [40].

Chen and Drucker simplified the analysis of Aaronson et al. in [3], providing a BellQMA protocol for $3SAT$ [22]. The BellQMA is a restricted class of QMA protocols in which the verifier is neither allowed to make entangled measurements nor condition them to previous outcomes.

Harrow and Montanaro showed that $\text{QMA}(k) = \text{QMA}(2)$ for k a polynomial in the input size [51]. This result is a corollary of a method called “product test”. Given the hypothesis of having two equal unentangled states, the product test succeeds with high probability if and only if these states are close to a product of k sub-states. In other words, given a bipartite unentanglement hypothesis, it is possible to verify k -partite entanglement. An important question is how difficult it is to verify k -partite entanglement without this hypothesis.

Chailloux and Sattath, using the ideas of Harrow and Montanaro for $\text{QMA}(k) = \text{QMA}(2)$, showed that the Separable Sparse Hamiltonian problem is $\text{QMA}(2)$ -complete whereas the Separable Local Hamiltonian problem is still only QMA-complete [21]. Moreover, two other problems were shown to be $\text{QMA}(2)$ -complete, namely, QPROD-ISOMETRY and QSEP-ISOMETRY. These problems ask for an input to the isometry such that the output is close to a product and a separable state, respectively. Furthermore, an explicit algorithm for $\text{QMA}(2)$ achieving better than a NEXP upper bound for special cases was given in [90] [37].

Using a new quantum de Finetti Theorem, Brandão et al. showed that for the special case of parallel one-way LOCC ($\text{LOCC}_1^{\parallel}$) verifiers, it holds that $\text{QMA}(k)_{\text{LOCC}_1^{\parallel}} = \text{QMA}$ for $k \in O(1)$ [18]. In a k party system, the parallel one-way LOCC class of measurements operationally works as follows: the first $k - 1$ parties apply each a POVM independently, and the last system is measured conditioned on the previous outcomes. Latter, this result was extended to show that $\text{BellQMA}(k) = \text{QMA}$ for a polynomial k [19]. An analogous result was shown for the fully one way LOCC class (LOCC_1) in which each party is measured sequentially with a POVM defined by previous outcomes [68]. This result led to a more general collapse, one in which we have $\text{QMA}(k)_{\text{LOCC}_1} = \text{QMA}$ for a polynomial k in the input.

Pereszlényi analyzed the case of QMA(2) with small gaps, in which small means at most inverse exponential in the input size [77]. Using the same ideas of Blier and Tapp [17] and a NEXP-complete problem, he devised a protocol to show that NEXP is equal to QMA(2) with a small gap. Assuming $\text{EXP} \neq \text{NEXP}$ and using the fact that QMA with a small gap is contained in EXP, we have a separation between QMA and QMA(2) in this small gap context. Combining our generic QMA(2) protocol and the classical PCP verifier in the proof of $\text{NEXP} \subseteq \text{PCP}(\text{poly}(n), O(1))$, we can achieve a similar result.

Apart from the collapse $\text{QMA}(k) = \text{QMA}(2)$, there are many open questions regarding the power of QMA(2). For the general case, a lower bound better than the trivial QMA is not known. Neither is an upper bound better than the trivial NEXP one. Watrous and Marriott showed that $\text{QMA}(1)_{\log(n)} \subseteq \text{BQP}$ [69], but this proof was not enough to show $\text{QMA}(2) = \text{QMA}$.

8.3 Preliminaries

In this subsection, we define complexity classes and state a few theorems which are relevant to this chapter.

8.3.1 Proof Verification

Many different characterizations of NP have been proposed in terms of PCPs. What makes this probabilistic characterization so attractive is that it was shown that NP is contained in $\text{PCP}_{1, \frac{1}{2}}(O(\log(n)), O(1))$. This means that it is possible to rewrite any witness of a NP instance in such a way that a probabilistic verifier only queries a constant number of positions [11] [73]. The following is a specific characterization of the NP-complete problem 3SAT.

Theorem 8.3.1 (Theorem 7 from [73]). *There exist a constant $\delta > 0$ and an alphabet Σ of constant size such that $3\text{SAT} \in \text{PCP}_{1, 1-\delta}(O(\log(n)), 2)$. Moreover, the PCP verifier makes two query projection tests, and the proof size m is almost linear, that is $m = n^{1+o(1)}$.*

Note that there are equivalent formulations of PCPs in terms of constraint satisfaction problems and multi-player games [29] [10]. Here, we focus our attention on the proof verification characterization.

There is also a similar result for the scaled up analogue of NP, that is, NEXP. The PCP verifier for NEXP also queries only a constant number of bits as stated by the next Theorem.

Theorem 8.3.2 (Adapted from Theorem 2.7 of [70]).

$$\text{NEXP} \subseteq \text{PCP}_{1, \frac{1}{2}}(O(\text{poly}(n)), O(1))_{\Sigma},$$

where $\Sigma = \{0, 1\}$ and the verifier runs in $\text{poly}(n)$ time.

8.3.2 Trace Distance

In the following, we make use of an operational property of the trace distance 2.6.7. This distance is equal to the optimal bias in distinguishing two quantum states with a promise that we are given one of them uniformly at random. For instance, let P and Q be the probability distribution when measuring these states in the computational basis, then the statistical distance $D(P, Q)$ can be used as a lower bound for the trace distance.

8.4 Quantum multiprover protocol for PCPs

In this section, we show a $\text{QMA}(q(n))$ protocol that simulates a classical $\text{PCP}_{1, 1-\epsilon_0}(r(n), q(n))_{\Sigma}$ verifier. The protocol is presented in detail next, followed by the proof of its completeness and soundness. It is an extension of Blier and Tapp's protocol [17]. Part of our analysis is similar to theirs, but it is also tighter and more general.

8.4.1 Protocol

The idea of our protocol is to encode the classical witness as a uniform superposition of indexes and their corresponding values. The quantum verifier expects a state in the form:

$$\sum_{i=1}^m \frac{1}{\sqrt{m}} |i\rangle |y_i\rangle, \quad (8.1)$$

where $y = y_1 \dots y_m$ is the original classical witness. The first and second parts of the quantum register are referred to as position and value registers, respectively. This witness is in the $\mathcal{H}_m \otimes \mathcal{H}_{|\Sigma|}$ Hilbert space, with total dimension $m|\Sigma|$.

The quantum verifier V is responsible for two tasks: verifying that the received proofs are close to a proper state of the form 8.1, and simulating the classical verifier for $\text{PCP}_{1, 1-\epsilon_0}(r(n), q(n))_{\Sigma}$. We refer to this verifier as V_0 . Note that using $r(n)$ random bits and making $q(n)$ queries, it is not possible to query more than $q(n)2^{r(n)}$ different positions in a witness. Therefore, the proof size can be bounded by $\lceil \log(|\Sigma|) \rceil q(n)2^{r(n)}$. Since this proof is represented as a uniform superposition in the quantum case, its size has an upper bound of $O(\lceil \log(|\Sigma|) \rceil + \log(q(n)2^{r(n)}))$, if represented by qubits. Moreover, it is possible to assume that $q(n)$ is at least two as additional queries can be ignored.

The quantum verifier conducts four tests in which the first three check if the state is close to a proper state, and the last one checks if the encoded classical witness is valid. More specifically, the first one is an equality test that ensures that the quantum states are close using the swap test. The second test certifies that all positions are present in the quantum proof. The third one ensures that the value of a given position is consistent in all proofs. Finally, the last test simulates V_0 . Each time V_0 tries to read a position in y , V measures one of its unmeasured quantum proofs and the simulation continues if and only if the measured and queried positions are the same. Otherwise, V accepts the input in order to avoid losing perfect completeness.

A precise description of V is given next. With equal probability, one of the four tests is performed.

- **Test 1:** (equality of certificates) Choose randomly two out of the $q(n)$ proofs, and denote them by $|\phi\rangle$ and $|\psi\rangle$. Accept if and only if $\text{SWAP}(|\phi\rangle, |\psi\rangle)$ accepts.
- **Test 2:** (all nodes are present) For each quantum proof $|\psi\rangle$:
 - Apply the QFT to the value portion of $|\psi\rangle$, and measure it.
 - If the outcome is frequency zero, measure the position register in the Fourier basis. If the outcome is not $|\bar{0}\rangle$, then reject.

If all quantum proofs pass the previous test, then accept.

- **Test 3:** (positions have consistent values) Choose randomly two out of the $q(n)$ proofs, denote them by $|\phi\rangle$ and $|\psi\rangle$. Measure these proofs. If the same position was retrieved in both, accept if and only if both have the same value.
- **Test 4:** (check using V_0) Simulate V_0 . Each time V_0 attempts to query a position, measure an unmeasured quantum proof. If the queried position and the measured position are the same, continue the simulation with the associated value. Otherwise accept, in order to ensure perfect completeness. Suppose V_0 successfully retrieved the values of all queried positions. In this case, accept if and only if V_0 accepts.

8.4.2 Completeness

We will now prove that the protocol described in the last subsection accepts with probability 1.

Let $y = y_1 y_2 \dots y_m$ be the classical $\text{PCP}_{1,1-\epsilon}(r(n), q(n))_\Sigma$ proof. The $q(n)$ optimal $\text{QMA}(q(n))_{\log(m)}$ proofs will be equal and encoded in a proper state of the form 8.1. We will now prove that each test accepts with probability 1.

Since all proofs are identical pure states, when we apply the swap test its failure probability is zero. Therefore, test 1 always accepts.

Each position has a well defined value $|y_i\rangle$ that is $\beta_{ij} = 1$ for some $j \in [|\Sigma|]$. In this case, after a QFT on the value register is performed, if a frequency of zero is measured, we know that the position register is in the superposition $\frac{1}{\sqrt{m}} \sum_{i=1}^m |i\rangle$. Consequently, test 2 accepts with probability 1. Furthermore, test 3 never rejects these certificates as values are consistent in all proofs.

Finally, if test 4 fails to retrieve a queried position, it accepts to ensure perfect completeness. Otherwise, if all positions were correctly retrieved, it can rely on the simulation of the classical PCP verifier that itself has perfect completeness.

8.4.3 Soundness

To prove the soundness of V , a series of lemmas are required. Despite the fact that the classical PCP verifier is robust for instances not in the language, the main concern in the quantum case is to ensure that it will be able to query the desired positions with non zero probability. Otherwise, our protocol would always accept.

Now, we briefly outline a sequence of lemmas and their respective goals. Firstly, in Lemma 8.4.1, by establishing a maximum failure probability in test 1, it is possible to conclude that the amplitudes in the proofs are component-wise close. Lemma 8.4.2 uses the previous Lemma and a probability bound for test 3, to conclude that, for positions with high amplitude, their values are well defined, *i.e.*, have a high amplitude. Using some positions with well defined values, the QFT in the value register succeeds with non zero probability, as shown by Lemma 8.4.3.

In a generic way, Lemma 8.4.4 shows that a quantum state cannot differ much from a uniform superposition, if we require state $|\bar{0}\rangle$ to be measured in the Fourier basis. Combining Lemmas 8.4.3 and 8.4.4, Lemma 8.4.5 establishes a minimum amplitude for all positions. Finally, assuming tests 1 to 3 succeed with at least the probability stated in the previous Lemmas, it is possible to determine the total rejecting probability of test 4, which uses the classical PCP verifier.

As some of the lemmas depend on the assumptions of previous lemmas, to simplify their description we just mention this dependency where necessary. Further, we establish a common notation when writing quantum proofs. In tests 1 to 3, out of the $q(n)$ proofs, we work with at most two of them. These two proofs denoted, by $|\psi\rangle$ and $|\phi\rangle$, are defined

as follows, in terms of their amplitudes:

$$\begin{aligned} |\psi\rangle &= \sum_i \alpha_i |i\rangle \sum_j \beta_{ij} |j\rangle, \quad \text{and} \\ |\phi\rangle &= \sum_i \alpha'_i |i\rangle \sum_j \beta'_{ij} |j\rangle, \end{aligned}$$

where $\sum_i |\alpha_i|^2 = \sum_i |\alpha'_i|^2 = 1$, and for all i it holds that $\sum_j |\beta_{ij}|^2 = \sum_j |\beta'_{ij}|^2 = 1$.

The first part of the quantum register is used to store the position and the second part stores the value from the alphabet Σ . Moreover, the probability of choosing a specific pair of proofs is $\frac{1}{\binom{q(n)}{2}}$. Since the latter appears in several of the following lemmas, we will denote it simply by p_{pair} .

The following Lemma states that if test 1 (the swap test) succeeds with high probability, then the quantum proofs are component-wise close.

Lemma 8.4.1. *If there exists k and l such that $||\alpha_k \beta_{kl}|^2 - |\alpha'_k \beta'_{kl}|^2| \geq \frac{1}{f_{swap}(m)}$, where $f_{swap}(m) \in \Omega(m)$, then test 1 fails with probability at least $\frac{p_{pair}}{8f_{swap}(m)^2}$.*

Proof. Let $P_{i,j} = |\alpha_i \beta_{ij}|^2$ and $Q_{i,j} = |\alpha'_i \beta'_{ij}|^2$ be the probability distributions when $|\psi\rangle$ and $|\phi\rangle$ are measured in the computational basis. Using the properties of the trace distance $D(|\psi\rangle, |\phi\rangle)$, it is possible to find an upper bound on $|\langle\psi|\phi\rangle|^2$, as shown next:

$$\begin{aligned} \sqrt{1 - |\langle\psi|\phi\rangle|^2} &= D(|\psi\rangle, |\phi\rangle) \\ &\geq D(P, Q) \\ &= \frac{1}{2} \sum_{ij} ||\alpha_i \beta_{ij}|^2 - |\alpha'_i \beta'_{ij}|^2| \\ &\geq \frac{1}{2} ||\alpha_k \beta_{kl}|^2 - |\alpha'_k \beta'_{kl}|^2| \\ &= \frac{1}{2} \frac{1}{f_{swap}(m)}. \end{aligned}$$

With the bound $|\langle\psi|\phi\rangle|^2 \leq 1 - \frac{1}{4f_{swap}(m)^2}$, we can obtain a lower bound on the swap test failure probability, that is,

$$\frac{1}{2} - \frac{|\langle\psi|\phi\rangle|^2}{2} \geq \frac{1}{8f_{swap}(m)^2}.$$

Since the probability of selecting a specific pair of proofs is p_{pair} , the total failure probability of test 1 is at least

$$\frac{p_{pair}}{8f_{swap}(m)^2}.$$

□

The next Lemma states that positions with high amplitude have well defined values. One important difference from Blier and Tapp's work is the extension to arbitrary alphabets [17].

Lemma 8.4.2. *If test 1 fails with probability $p_{fail_1} < \frac{p_{pair}}{8f_{swap}(m)^2}$, with $f_{swap}(m) = 4m|\Sigma|$, and test 3 fails with probability $p_{fail_3} < p_0 p_{pair} p_r^2$, then for any position l such that $|\alpha_l|^2 \geq \frac{1}{2m}$ the following hold:*

- (i) *position l in $|\phi\rangle$ has $|\alpha'_l|^2 > \frac{1}{4m|\Sigma|}$;*
- (ii) *position l can be measured in $|\psi\rangle$ or $|\phi\rangle$ with probability at least $p_r = \frac{1}{4m|\Sigma|}$;*
- (iii) *for any $K \in]0.5, 1[$, there is a value j in $[\Sigma]$ such that both $|\beta_{lj}|^2, |\beta'_{lj}|^2 \geq K$, given the right choice of $p_0 \in]0, 0.5[$.*

Proof.

- (i) From Lemma 8.4.1, test 1 guarantees that $||\alpha_l \beta_{lj}|^2 - |\alpha'_l \beta'_{lj}|^2| < \frac{1}{f_{swap}(m)}$ when $p_{fail_1} < \frac{p_{pair}}{8f_{swap}(m)^2}$. Further, there is a j such that $|\beta_{lj}|^2 \geq \frac{1}{|\Sigma|}$ since $\sum_j |\beta_{lj}|^2 = 1$. For this j , the term $|\alpha_l \beta_{lj}|^2$ can be bounded below by $\frac{1}{2m|\Sigma|}$, giving

$$|\frac{1}{2m|\Sigma|} - |\alpha'_l|^2 |\beta'_{lj}|^2| < \frac{1}{f_{swap}(m)}.$$

Since $|\beta'_{lj}|^2$ is at most 1, we have $|\alpha'_l|^2 > \frac{1}{2m|\Sigma|} - \frac{1}{f_{swap}(m)}$. Taking $f_{swap}(m)$ as $4m|\Sigma|$, a lower bound for $|\alpha'_l|^2$ becomes

$$|\alpha'_l|^2 > \frac{1}{2m|\Sigma|} - \frac{1}{f_{swap}(m)} > \frac{1}{2m|\Sigma|} - \frac{1}{4m|\Sigma|} = \frac{1}{4m|\Sigma|}.$$

- (ii) To retrieve position l in a quantum proof, it is possible to measure this proof directly. From item (i), this measurement succeeds with probability at least $p_r = \frac{1}{4m|\Sigma|}$, for both proofs $|\psi\rangle$ and $|\phi\rangle$.
- (iii) By adjusting p_0 , in the upper bound of the failure probability of test 3, given by $p_{fail_3} < p_0 p_{pair} p_r^2$, it is possible to enforce that position l has one value with high amplitude. Let $x = [|\beta_{l1}|^2, \dots, |\beta_{l|\Sigma|}|^2]$ and $y = [|\beta'_{l1}|^2, \dots, |\beta'_{l|\Sigma|}|^2]$ denote two vectors containing the squared amplitudes of the register values for positions l of $|\psi\rangle$ and $|\phi\rangle$, respectively. Suppose that position l was measured in both proofs. We denote by p_0 the probability of detecting if the content of register values disagree, in which case test 3 rejects. This probability can be written as

$$p_0 = \sum_{j \neq k} |\beta_{lj}|^2 |\beta'_{lk}|^2.$$

If $p_{fail_3} < p_0 p_{pair} p_r^2$, we have $\sum_{j \neq k} |\beta_{lj}|^2 |\beta'_{lk}|^2 < p_0$, or conversely

$$\sum_j |\beta_{lj}|^2 |\beta'_{lj}|^2 \geq 1 - p_0.$$

The previous sum can be rewritten as the inner product

$$\langle x | y \rangle \geq 1 - p_0,$$

where the two vectors are in $\mathbb{R}^{|\Sigma|}$, $\|x\|_1 = \|y\|_1 = 1$, and their components are non-negative.

For $p_0 < 0.5$, Lemma A.1.1 can be applied, showing the existence of a j such that $|\beta_{lj}|^2$ and $|\beta'_{lj}|^2$ are greater or equal than any fixed $K \in]0.5, 1[$, given the appropriate choice of p_0 . \square

By bounding the success probability of tests 1 and 3, it is possible to bound the probability of retrieving a frequency of zero in the register value after a QFT is performed.

Lemma 8.4.3. *Given the assumptions of Lemma 8.4.2, the probability of measuring a frequency of zero in the register value, after a QFT is applied to it, is greater than $\frac{1}{3|\Sigma|}$.*

Proof. If position i was measured, the probability of measuring frequency zero in the register value is

$$\frac{1}{|\Sigma|} |\beta_{i0} + \dots + \beta_{i|\Sigma|}|^2.$$

From Lemma 8.4.2, we know that there is an i (we renamed l to i) and a j such that $|\alpha_i|^2 \geq \frac{1}{2m}$ and $|\beta_{ij}|^2 \geq K$. Without loss of generality let $j = 1$. We have

$$\frac{1}{|\Sigma|} |\beta_{i1} + \dots + \beta_{i|\Sigma|}|^2 \geq \frac{1}{|\Sigma|} (|\beta_{i1}| - |\sum_{j=2}^{|\Sigma|} \beta_{ij}|)^2.$$

By appropriately choosing K , the factor $\frac{2}{3|\Sigma|}$ can be used as a lower bound for the previous expression.

At most $m - 1$ positions may satisfy $|\alpha_i|^2 \leq \frac{1}{2m}$. As a result, the probability of obtaining a frequency of zero is at least

$$(1 - (m - 1) \frac{1}{2m}) \frac{2}{3|\Sigma|} \geq \frac{1}{3|\Sigma|}.$$

This completes the proof. \square

For a generic quantum state, if we intend to measure $|\bar{0}\rangle$ in the Fourier basis with high probability, then Lemma 8.4.4 gives a minimum amplitude for each position.

Lemma 8.4.4. *Given a state $|\psi\rangle = \sum_i \gamma_i |i\rangle$, and a l such that $|\gamma_l|^2 < \frac{1}{2m}$, the probability of not getting $|\bar{0}\rangle = F_m|0\rangle$ when $|\psi\rangle$ is measured in the Fourier basis is at least $\frac{1}{16m^2}$.*

Proof. Let P and Q be the probability distributions when measuring $|\psi\rangle$ and $|\bar{0}\rangle$ in the computational basis, respectively. Using the properties of the trace distance, we have

$$\begin{aligned} \sqrt{1 - |\langle\psi|\bar{0}\rangle|^2} &= D(|\psi\rangle, |\bar{0}\rangle) \\ &\geq D(P, Q) \\ &= \frac{1}{2} \sum_i ||\gamma_i|^2 - \frac{1}{m}| \\ &\geq \frac{1}{2} ||\gamma_l|^2 - \frac{1}{m}| \\ &\geq \frac{1}{4m}. \end{aligned}$$

The value $1 - |\langle\psi|\bar{0}\rangle|^2$ is the probability of not getting $|\bar{0}\rangle$ when measuring in the Fourier basis. In this case, it is at least $\frac{1}{16m^2}$. \square

Lemma 8.4.5. *If the assumption of Lemma 8.4.3 is met, and test 2 fails with probability $p_{fail_2} < \frac{1}{48|\Sigma|m^2}$, then $|\alpha_i|^2 \geq \frac{1}{2m}$, for all i .*

Proof. With the assumption of Lemma 8.4.3, there is a $\frac{1}{3|\Sigma|}$ probability of measuring a frequency of zero for the register value. In that case, Lemma 8.4.4 implies that, for all i , $|\alpha_i|^2 \geq \frac{1}{2m}$ or, otherwise, $p_{fail_2} \geq \frac{1}{3|\Sigma|} \frac{1}{16m^2} = \frac{1}{48|\Sigma|m^2}$. \square

Lemma 8.4.6. *Let V_0 be the classical PCP verifier, and let $1 - \epsilon_0$ be its soundness. If the assumptions of Lemma 8.4.5 are met, then test 4 rejects the input x with probability at least $\epsilon_0(K\frac{1}{2m})^{q(n)}$ when x is not in the language of V_0 .*

Proof. When the hypothesis of Lemma 8.4.5 are met, each position has probability at least $\frac{1}{2m}$. In this case, item (iii) of Lemma 8.4.2 also applies. Thus each position has a well defined value with probability K . The encoded classical witness is considered to be composed of these values.

The probability of correctly measuring each of the $q(n)$ positions desired by the verifier V_0 , is at least

$$(K\frac{1}{2m})^{q(n)}.$$

Consequently, the total probability of detecting a no-instance is at least

$$\epsilon_0(K \frac{1}{2m})^{q(n)}.$$

The proof is complete. \square

Now we are ready to state our main result showing the relationship between a generic classical PCP verifier and the complexity class QMA(k).

Theorem 8.4.7.

$$\text{PCP}_{1,1-\epsilon_0}(r(n), q(n))_\Sigma \subseteq \text{QMA}(q(n), 1, 1 - \epsilon)_{l(n, \Sigma)},$$

where

- the classical witness size m is a function of n ;
- $l(n, \Sigma) = O(\lceil \log(|\Sigma|) \rceil + \log(m))$;
- $\epsilon = \min\{\frac{p_{\text{pair}}}{km^2}, \epsilon_0(K \frac{1}{2m})^{q(n)}\}$;
- $k = \frac{48|\Sigma|^2}{p_0}$;
- $p_0 \in]0, 0.5[$;
- $K \in]0.5, 1[$ is a function of p_0 ;
- $p_{\text{pair}} = \frac{1}{\binom{q(n)}{2}}$.

Proof. Firstly, we observe that claim A.1.2 allows the encoding of the classical $\text{PCP}_{1,1-\epsilon_0}(r(n), q(n))_\Sigma$ witness using qubits instead of qudits. The conversion process at most doubles the witness and alphabet sizes, while all other parameters remain the same. This observation is important because the Fourier unitary operator will always admit an efficient implementation, since the position and register values have at most polynomially many qubits in n .

To prove the Theorem we use the protocol described at the beginning of Section 8.4.1. A proof of its perfect completeness was already given in Section 8.4.2, and it is only left to show its soundness, which also follows by combining the previous lemmas.

Tests 1 to 3 of the quantum verifier V ensure that the proofs are well formed in the sense that they have a minimum amplitude for all positions, and the associated values are consistent in all proofs. Assuming the probability of failure in each of these tests are smaller than $\frac{p_{\text{pair}}}{km^2}$, for $k = \frac{48|\Sigma|^2}{p_0}$, we have a series of implications that makes the hypothesis of Lemma 8.4.6 hold. Lemma 8.4.2 implies the statement in Lemma 8.4.4, which in turn implies the statement in Lemma 8.4.5. Since the hypothesis of Lemma 8.4.6 is met, test 4 fails with probability at least $\epsilon_0(K \frac{1}{2m})^{q(n)}$. As each test is selected with equal probability, the soundness is at most $1 - \min\{\frac{p_{\text{pair}}}{km^2}, \epsilon_0(K \frac{1}{2m})^{q(n)}\}$. \square

As a corollary, PCP verifiers can be simulated with two unentangled provers.

Corollary 8.4.8.

$$\text{PCP}_{1,1-\epsilon_0}(r(n), q(n))_\Sigma \subseteq \text{QMA}(2, 1, 1 - \epsilon')_{l'(n, \Sigma)}.$$

Proof. Follows directly from Theorem 8.4.7, and from $\text{QMA}(k) = \text{QMA}(2)$ in Lemma 4.7.2. The parameters are the same as in the previous theorem, and we let $\epsilon' = \frac{\epsilon^2}{100}$. \square

For the specific case of the language $3SAT$, we have the following corollary.

Corollary 8.4.9. *Language $3SAT$ is in $\text{QMA}(2, 1, 1 - \Omega(\frac{1}{n^{2+\varepsilon}}))_{\log(n)}$, for any $\varepsilon > 0$.*

Proof. Theorem 8.3.1 states that $3SAT$ is in $\text{PCP}_{1,1-\delta}(O(\log(n)), 2)_\Sigma$ for constants δ and $|\Sigma|$. Also, the proof size is $m = n^{1+o(1)}$. From Theorem 8.4.7, we conclude that $3SAT$ is in $\text{QMA}(2, 1, 1 - \Omega(\frac{1}{n^{2+\varepsilon}}))_{\log(n)}$ for any $\varepsilon > 0$. \square

With our generalization, it is possible to obtain again the known result $\text{NEXP} = \text{QMA}(2)$ with a small gap[77]. But now without the need of going into the details of a specific complete problem for NEXP .

Corollary 8.4.10.

$$\text{NEXP} = \text{QMA}(2, 1, 1 - \Omega(\frac{1}{2^{\text{poly}(n)}}))_{\text{poly}(n)}.$$

Proof. The containment $\text{QMA}(2, 1, 1 - \Omega(\frac{1}{2^{\text{poly}(n)}}))_{\text{poly}(n)} \subseteq \text{NEXP}$ is straightforward. An EXP time classical verifier receives the two exponential size quantum witnesses (up to a doubly exponential precision) from a prover and then it simulates the quantum verifier.

The other containment $\text{NEXP} \subseteq \text{QMA}(2, 1, 1 - \Omega(\frac{1}{2^{\text{poly}(n)}}))_{\text{poly}(n)}$ follows by combining Theorem 8.3.2 and Corollary 8.4.8 in a similar way, as in the previous proof. \square

Note 8.4.11. *From this last corollary, it is possible to conclude a known result, namely, optimizing the function $\text{Tr}(M\rho^{AB})$ over the separable states is NP-hard up to an inverse polynomial error in the dimension, where $0 \leq M \leq I$ [20]. This is one of the reasons why it is challenging to find an upper bound for $\text{QMA}(2)$ better than NEXP .*

8.5 Discussion

In this work, we generalized the Blier and Tapp protocol to simulate any classical PCP verifier. The quantum setting allows an exponential reduction in the proof size, while also reducing the completeness-soundness gap of the system. With our generalization and two concrete classical PCP Theorems, we recover two known results. Firstly, using a PCP Theorem for $3SAT$, we find a completeness-soundness gap of $\Omega(\frac{1}{n^{2+\varepsilon}})$ for any $\varepsilon > 0$ which lies between the Beigi and the Le Gall et al. results. Moreover, with a PCP Theorem for NEXP, we recover that QMA(2) is equal to NEXP in the context of small gaps, as in Pereszlényi [77].

We ask if it is possible to adapt a classical PCP verifier to take advantage of the probability distribution naturally associated with quantum amplitudes. That is, instead of looking for specific positions in the proof, the verifier might enforce some minimum amplitude for each position, or some other distribution, in an attempt to increase the completeness-soundness gap. Another question is if it is possible to use the quantum setting developed here to derive classical PCP hardness results. For instance, it might be possible to obtain a lower bound for the number of queries, given a certain alphabet size, based on the assumption that $P \neq NP$.

Chapter 9

On the Hardness of Disentanglers and Quantum de Finetti Theorems

9.1 Introduction

Entanglement is an important resource in quantum information processing. It is a fundamental ingredient in protocols such as teleportation and superdense coding [98]. Nonetheless, there are situations in which the lack of entanglement may lead to better resource usage. This duality becomes evident in the context of Multi-prover Quantum Merlin-Arthur complexity classes, denoted $\text{QMA}(k)$, where k is the number of unentangled provers [65]. We know that there are a variety of protocols in $\text{QMA}(2)$ for treating NP-complete problems, which use only a logarithmic number of qubits with respect to the input size [17] [14] [23] [40]. But unentanglement promises seem hard to enforce both quantumly and classically. For instance, the expressive power of $\text{QMA}(2)$ is not yet well understood, as it ranges from the QMA class [63] to the powerful NEXP class [10]. In this chapter, we investigate the potential limits of approaches for breaking and controlling entanglement. We do that by making use of some computational hardness results.

A quantum state that is not entangled is said to be separable. In the case of a mixed state σ^{AB} for two subsystems A and B , it is separable if it can be written as $\sigma^{AB} = \sum p_i \sigma_i^A \otimes \sigma_i^B$, where p_i is a probability distribution [56]. Even though the set of separable states forms a convex set, finding the state that maximizes the Hilbert-Schmidt inner product of a positive semi-definite matrix M and separable state σ^{AB} is NP-hard when the error is an inverse polynomial in the input [42]. This is known as the Best Separable State (BSS) problem, and it is closely related to the Weak Membership Problem for a set of separable states [20]. Since quantum states with polynomially many qubits are objects of Hilbert spaces of exponential size, optimizing the classical description of separable quantum states may easily become a NEXP-hard problem.

Instead of dealing with the classical description of quantum states, an alternative approach would be to explore quantum ways of breaking entanglement. Aaronson et al. [3] were the first to propose a disentangler which is a quantum channel capable of approximating any separable state within an error δ in the trace norm, and with output guaranteed to be always ϵ -close to a separable state, also in the trace norm. They proved that there is no perfect disentangler when the errors δ and ϵ are set to zero. We extend their disentangler definition by considering the computational complexity of the quantum channel as well as the relationship of the input and output space dimensions. This will allow us to associate computational hardness results to the existence of some disentanglers.

Computational hardness results are useful to unveil how hard it is to solve a certain problem by implying that its solution would also solve problems known, or believed, to be hard. The NP-hardness is perhaps the most important example, and problems in this class are widely believed to be intractable [41]. Another classical hardness class stems from the hypothesis that *3SAT* has no sub-exponential time algorithm, also known as the Exponential Time Hypothesis (ETH) [55]. Here, we explore two quantum hardness notions: the hypothesis that *3SAT* can not be solved in quantum polynomial time (the BQP class) and the belief that the best quantum algorithm for this problem requires $\Omega(2^{\sqrt{n}})$ time.

A different way to understand and control entanglement is through quantum de Finetti Theorems. Given a permutation invariant state $\rho_{A_1 \dots A_n}$ on n subsystems of dimension $|A|$, a generic de Finetti Theorem bounds the distance of a reduced state $\rho_{A_1 \dots A_k}$ on k ($k \leq n$) subsystems and a separable state. This distance is usually a function of k , n and $|A|$. Moreover, this distance can be measured using different norms such as the standard trace norm, the **SEP** norm, or the fully one-way **LOCC** norm [51] [68]. Using the connection of disentanglers with de Finetti Theorems, it is possible to establish a hardness result on how the error scales with the number n in the **SEP** norm. Given the hardness assumption that *3SAT* requires $\Omega(2^{\sqrt{n}})$ quantum time, this distance decreases at best as an inverse polynomial in n .

9.2 Preliminaries

A disentangler is a quantum channel capable of breaking entanglement. It has numerous applications in quantum information, quantum computing and quantum complexity. For instance, the existence of a certain efficient disentangler can be used to show the collapse $\text{QMA}(2) = \text{QMA}$. We extend the disentangler definition of Aaronson et al., to take into account the input and output space dimensions, as given next.

Definition 9.2.1 (Adapted from [3]). *Let \mathcal{H} and \mathcal{K} be two finite-dimensional Hilbert*

spaces. Then given a super-operator $\Phi : \mathcal{H} \rightarrow \mathcal{K} \otimes \mathcal{K}$, we say Φ is an (ϵ, δ, f) -disentangler if

- (i) $\Phi(\rho)$ is ϵ -close to a separable state for every ρ ,
- (ii) for every separable state σ , there exists a $\rho \in \mathcal{H}$ such that $\Phi(\rho)$ is δ -close to σ , and
- (iii) $\log(\dim(\mathcal{H})) = f(\log(\dim(\mathcal{K})))$, where $f : \mathbb{R}^+ \rightarrow \mathbb{R}^+$.

In the previous definition, closeness is measured with respect to the trace distance. However, it is also possible to use distances based on other norms such as **SEP** and the fully one-way **LOCC**. We denote by Φ^M , the restriction of Φ when the distance is measured according to the measurement class M .

We briefly describe three measurement classes: parallel one-way LOCC ($\mathbf{LOCC}_1^\parallel$), **LOCC** and **SEP**, as defined in [51]. They describe the allowed measurement operators that are sometimes operationally motivated, such as $\mathbf{LOCC}_1^\parallel$ and **LOCC**. The $\mathbf{LOCC}_1^\parallel$ class comprises all measurements that can be performed first on a subsystem A , and according to its outcome, an appropriate measurement M_i is used on a subsystem B . Such an operator can be written as:

$$M = \sum_i \alpha_i \otimes M_i,$$

where $\{\alpha_i\}$ forms a positive operator-valued measure (POVM) and $0 \leq M_i \leq I$ for each i . The more general class **LOCC** comprises measurements on subsystems that can be implemented using a finite number of local measurements and classical communication. In terms of operators, it can be inductively described as

$$\begin{aligned} M &= \sum_i (\sqrt{E_i} \otimes I) M_i (\sqrt{E_i} \otimes I), \quad \text{or} \\ M &= \sum_i (I \otimes \sqrt{E_i}) M_i (I \otimes \sqrt{E_i}), \end{aligned}$$

where $\{E_i\}$ satisfies $\sum_i E_i \leq I$ and $\{M_i\} \in \mathbf{LOCC}$. The **SEP** operator is even more expressive. It can be defined as

$$M = \sum_i M_i^A \otimes M_i^B,$$

for positive semi-definite rank one matrices M_i^A and M_i^B . We have the following inclusion of measurement classes

$$\mathbf{LOCC}_1^\parallel \subseteq \mathbf{LOCC} \subseteq \mathbf{SEP}.$$

As in [68], it is possible to associate with every POVM $\{M_x\}$ and state ρ the new state $\mathcal{M}(\rho) = \sum_x \text{Tr}(M_x \rho) |x\rangle\langle x|$. Using this definition, it is possible to write the norm of a class of operator M as

$$\|\rho - \sigma\|_M = \max_{\mathcal{M} \in M} \|\mathcal{M}(\rho) - \mathcal{M}(\sigma)\|_1.$$

Note that this norm has an operational interpretation as the optimum bias that can be achieved by an operator of the class M when distinguishing ρ and σ , given one of them with uniform probability. This is the same operational interpretation of the trace norm [74].

The action of a quantum channel Φ on a state ρ^A can be described by a unitary operator U^{AE} , acting on it and the environment, which is initialized with $|0\rangle\langle 0|^E$, followed by tracing out all subsystems in AE but for a subset B . This action is described as

$$\sigma^B = \Phi(\rho^A) = \text{Tr}_{\setminus B}(U^{AE} \rho^A \otimes |0\rangle\langle 0|^E U^{AE\dagger}).$$

One important observation is that the environment can always be modeled with quadratically many qubits as in system A [74].

We measure the time complexity of a channel Φ using the implementation of U^{AE} in the same way that we measure the complexity of quantum circuits by counting the number of elementary gate operations it uses, taken from a fixed universal gate set such as $\{\mathbf{H}, \mathbf{T}, \mathbf{CNOT}\}$ [64].

It is possible to restrict item (ii) in Definition 9.2.1 to apply only to states σ with certain properties, instead of to arbitrary separable states. One important class of separable states is given by

$$\sigma = \int \psi \otimes \psi d\mu(\psi),$$

where μ is a probability measure over density matrices of a given size. This type of state arises in several quantum de Finetti Theorems [24] [97]. The restriction of a disentangler Φ to require only approximation to states of this form in item (ii) is denoted $\Phi_{=}$.

We make use of the following amplification theorem of Watrous and Marriott, in which QMA with an inverse polynomial completeness soundness gap can be amplified to an exponentially small error using the same witness.

Theorem 9.2.2 ([69]). *Let $c, s : \mathbb{N} \rightarrow [0, 1]$, and $g \in \text{poly}$ with*

$$c(n) - b(n) \geq \frac{1}{g(n)},$$

for all $n \in \mathbb{N}$. Then $\text{QMA}(1, c(n), s(n))_m \subseteq \text{QMA}(1, 1 - 2^{-r(n)}, 2^{-r(n)})_m$ for every m and $r \in \text{poly}$. Moreover, the proof size m remains unchanged in the amplification.

Harrow and Montanaro showed that $\text{QMA}(k)$ collapses to $\text{QMA}(2)$.

Lemma 9.2.3 (From [51]). *For any $m, k \in \mathbb{N}$ and $0 \leq s < c \leq 1$,*

$$\text{QMA}(k, c, s)_m \subseteq \text{QMA}_{km}(2, c', s')^{\text{SEP}}$$

where $c' = \frac{1+c}{2}$ and $s' = 1 - \frac{(1-s)^2}{100}$. Further, for any language L in $\text{QMA}_{km}(2, c', s')^{\text{SEP}}$ and any input $x \in L$, the two witnesses may be considered equal without loss of generality.

9.3 Disentanglers

We know that there is no disentangler in which $\epsilon = \delta = 0$, from [3]. In this section we study the computational hardness when $\epsilon, \delta > 0$, and show that, under certain hardness assumptions, even allowing an exponential time channel with respect to the number of output qubits n_{out} , there is no disentangler that acts on $\text{poly}(n_{\text{out}})$ input qubits and has error $\epsilon = \delta < \frac{k}{2^{n_{\text{out}}}}$, for some constant k .

The next lemma shows how to use a (ϵ, δ, f) -disentangler with certain properties to guarantee $\text{QMA}(2) \subseteq \text{QMA}(1)_m$. Note that m suffers an increase that depends on f .

Lemma 9.3.1. *For functions $f, l : \mathbb{N} \rightarrow \mathbb{N}$ and $c, s : \mathbb{N} \rightarrow [0, 1]$ satisfying $c(n) - s(n) \geq \frac{1}{g(n)}$ where g is a polynomial, we have*

$$\text{QMA}(2, c(n), s(n))_{l(n)} \subseteq \text{QMA}(1)_{f(l(n))},$$

assuming there is a polynomial time (ϵ, δ, f) -disentangler Φ with $\epsilon = \delta \leq \frac{1}{4g(n)}$.

Proof. We show how to transform a $\text{QMA}(2, c(n), s(n))_{l(n)}$ verifier into a $\text{QMA}(1)_{f(l(n))}$ one. Let $L \in \text{QMA}(2, c(n), s(n))_{l(n)}$. If $x \in L$, there is a witness $|\psi\rangle = |\psi_1\rangle \otimes |\psi_2\rangle$ and a state ρ such that $\Phi(\rho) = \psi'$ is δ -close to this witness. In the $\text{QMA}(1)_{f(l(n))}$ protocol, the prover sends this state ρ with $f(l(n))$ qubits. The verifier applies the disentangler obtaining the approximation ψ' with which the original $\text{QMA}(2)$ protocol is executed. It is clear that completeness is at least $c(n) - \delta$ since $\Phi(\rho)$ is δ -close to $|\psi\rangle$. Otherwise, if $x \notin L$, no matter the state ρ sent by the prover, it will be ϵ -close to a separable state, making the final soundness be at most $s(n) + \epsilon$. The completeness soundness gap becomes

$$c(n) - s(n) - \delta - \epsilon \geq \frac{1}{g(n)} - \frac{1}{2g(n)} = \frac{1}{2g(n)},$$

and that is still inversely polynomial, completing the proof. \square

A $\text{QMA}(1)_m$ protocol can be simulated in time $O(\text{poly}(n)2^{2m})$ where n is the input size. This result, adapted from Watrous and Marriott [69], plays a crucial role in our hardness results for some disentanglers. It is stated as follows.

Lemma 9.3.2. *Let $L \in \text{QMA}(1)_m$, and $x \in \{0,1\}^n$. Deciding if $x \in L$ can be done in $O(\text{poly}(n)2^{2m})$ quantum time.*

Proof. We review the proof that $\text{QMA}_{O(\log(n))} \subseteq \text{BQP}$ in [69]. We start with some notations. Let L be a language in QMA_m where m denotes the witness size. Let x be an input string and A_x be the associated verifier circuit acting on $k + m$ qubits, where k is the number of ancilla qubits, a polynomial in the size of the input. The amplification procedure of Theorem 9.2.2 allows us to assume, without loss of generality, that if $x \in L$ then there is a witness $|\psi\rangle$ that satisfies

$$\Pr[A_x \text{ accepts } |\psi\rangle] \geq 1 - 2^{-m-2},$$

and if $x \notin L$, for all states $|\psi\rangle$ we have

$$\Pr[A_x \text{ accepts } |\psi\rangle] \leq 2^{-m-2}.$$

We associate every x with a $2^m \times 2^m$ matrix Q_x as follows

$$Q_x = (I_m \otimes \langle 0^k |) A_x^\dagger \Pi_1 A_x (I_m \otimes | 0^k \rangle),$$

where Π_1 is the projector on the subspace that has the accepting qubit of A_x equal to 1. Note that Q_x corresponds to a positive semidefinite matrix of an efficient implementable measurement, as A_x is a polynomial time circuit.

The eigenvalues of Q_x are the associated acceptance probabilities of their respective eigenvectors. Therefore, if $x \in L$, then $\text{Tr}(Q_x) \geq 1 - 2^{-m-2} \geq \frac{3}{4}$ as there exist at least one state (eigenvector) that accepts with probability at least $1 - 2^{-m-2}$. Otherwise, all eigenvalues are at most 2^{-m-2} , resulting in $\text{Tr}(Q_x) \leq 2^m 2^{-m-2} \leq \frac{1}{4}$. It is possible to build a BQP circuit B that decides if $L \in \text{QMA}_m$ by applying the measurement Q_x to the totally mixed state on m qubits. In this case, the acceptance probability of B is

$$\Pr[B \text{ accepts}] = \text{Tr}(Q_x 2^{-m} I_m) = 2^{-m} \text{Tr}(Q_x).$$

The completeness soundness gap $g(n)$ is 2^{-m-1} . We can repeat B a certain number of times, say N , in order to amplify this gap. Using the Chernoff bound to achieve a constant error probability ϵ_c , the value N must satisfy

$$N \geq \frac{1}{g^2} \ln\left(\frac{1}{\sqrt{\epsilon_c}}\right),$$

resulting in a time complexity $O(\text{poly}(n)2^{2m})$, given that the QMA_m verifier runs in time $\text{poly}(n)$. \square

The $\text{QMA}(2)_{\log(n)}$ protocol for $3SAT$ that has the largest completeness soundness gap is due to Le Gall et al., it is our starting point to show the hardness of some disentanglers.

Theorem 9.3.3 (GNN [40]).

$$3SAT \in \text{QMA}(2, 1, 1 - \Omega(\frac{1}{n^{\text{polylog}(n)}}))_{O(\log(n))}.$$

Let $\frac{1}{g(n)}$ denote the completeness soundness gap. The proof size $l(n)$ satisfies

$$\log(tg(n)) \leq l(n) \leq t' \log(n),$$

for constants t, t' and $n > 1$.

For a (ϵ, δ, f) -disentangler Φ , if the errors ϵ and δ are sufficiently small and f does not require the input size to be much larger than the output, it will be possible to use the two previous lemmas to show that $3SAT$ can be decided faster than it is currently known. This is our approach to show the hardness of some disentanglers. The following three theorems make precise the conditions under which Φ promotes such speedups.

The next theorem covers the case in which the input and output of Φ are linearly related.

Theorem 9.3.4. *If $3SAT \notin \text{BQP}$, then there is no $O(\text{poly}(\dim(\mathcal{K})))$ time (ϵ, δ, f) -disentangler Φ with $f(x) = cx$ and $\epsilon, \delta \leq \frac{k}{\dim(\mathcal{K})}$, for any constant $c \geq 1$ and any fixed constant k .*

Proof. This proof follows from the GNN protocol for $3SAT$ in Theorem 9.3.8, and using Lemmas 9.3.1 and 9.3.2, as we elaborate next. We show the contrapositive, that is, if a $O(\text{poly}(\dim(\mathcal{K})))$ time (ϵ, δ, f) -disentangler Φ exists with $f(x) = cx$ and $\epsilon, \delta \leq \frac{k}{\dim(\mathcal{K})}$ for any $c \geq 1$ and a constant k that we specify latter, then $3SAT$ is in BQP .

Let $g(n) = c(n) - s(n)$ be the completeness soundness gap in the GNN protocol where n denotes the input size. In this protocol, the proof size $l(n)$ is greater than $\log(tg(n))$ for some constant t . Thus, the dimension $\dim(\mathcal{K})$ of the output space is at least $tg(n)$. Since the errors ϵ and δ of Φ are bounded above by $\frac{k}{\dim(\mathcal{K})}$, we can choose a constant k such that $\frac{k}{tg(n)} \leq \frac{1}{4g(n)}$.

Combining the existence of Φ and Lemma 9.3.1, we conclude that the GNN protocol is in $\text{QMA}_{ct' \log(n)}$ for some constant t' , where $t' \log(n)$ is an upper bound on the witness size in this protocol. Now, using Lemma 9.3.2 this protocol can be run in polynomial $O(\text{poly}(n)2^{2ct' \log(n)})$ quantum time, implying that $3SAT$ is in BQP . \square

The previous result can be improved under the assumption that there is no quasi-polynomial quantum time ($O(2^{\text{polylog}(n)})$) algorithm for $3SAT$. The best quantum algorithm to this date is the Grover unstructured search, with time complexity $O(2^{\sqrt{n}})$.

Theorem 9.3.5. *If there is no quantum algorithm for 3SAT which runs in $O(2^{\text{polylog}(n)})$ time, then there is no $O(\text{poly}(\dim(\mathcal{K})))$ time (ϵ, δ, f) -disentangler Φ with $f \in \text{poly}$ and $\epsilon, \delta \leq \frac{k}{\dim(\mathcal{K})}$, where k is a constant.*

Proof. Similar to the proof of Theorem 9.3.4. But now the simulation of the GNN protocol occurs in $\text{QMA}_{\text{polylog}(n)}$, as the number of input qubits and output qubits of Φ are related by a polynomial. Since the output dimension Φ remains the same as in the previous theorem, the same choice of k allows us to use Lemma 9.3.1 and claim that 3SAT is in $\text{QMA}_{\text{polylog}(n)}$. Using Lemma 9.3.2, we conclude that 3SAT that can be solved in $O(\text{poly}(n)2^{\text{polylog}(n)})$ time. \square

We can also conjecture that every quantum algorithm to solve any NP-complete problem requires $\Omega(2^{\sqrt{n}})$ time. In this case, the function f in Theorem 9.3.5 can be improved again, as shown by the next theorem.

Theorem 9.3.6. *If there is no quantum algorithm for 3SAT which runs in $o(2^{\sqrt{n}})$ time, then there is no $O(\text{poly}(\dim(\mathcal{K})))$ time (ϵ, δ, f) -disentangler Φ with $f(x) = 2^{\frac{x}{c}}$ and $\epsilon, \delta \leq \frac{k}{\dim(\mathcal{K})}$, for some constants c and k .*

Proof. Use the same reasoning as in the last two results. Let the proof size in the GNN protocol be bounded by $t' \log(n)$ for a constant t' . It suffices to choose a constant c such that $2^{\frac{t' \log(n)}{c}} < \frac{\sqrt{n}}{2}$ for every $n > 1$. \square

The previous theorem is a step towards proving the following conjecture.

Conjecture 9.3.7 (Watrous (from [3])). *For any constants $\epsilon, \delta < 1$, a (ϵ, δ) -disentangler will require $\dim(\mathcal{H}) = 2^{\Omega(\dim(\mathcal{K}))}$.*

It would be interesting if the previous theorem could be scaled down. That is, instead of a $\dim(\mathcal{K})$ dependence, it would show a $\text{polylog}(\dim(\mathcal{K}))$ dependence for both the disentangler complexity and the errors ϵ and δ .

It is possible to use the result $\text{QMA}(k) = \text{QMA}(2)$, given by Lemma 9.2.3, to show that the GNN protocol can be transformed into a **SEP** protocol by doubling the message size and squaring the completeness soundness gap.

Theorem 9.3.8 (SEP GNN (variation of [40])). *We have*

$$3\text{SAT} \in \text{QMA}(2, 1, 1 - \Omega(\frac{1}{n^2 \text{polylog}(n)}))^{\text{SEP}}_{O(\log(n))}.$$

If $g(n) \in \Omega(\frac{1}{n^2 \text{polylog}(n)})$ is the completeness soundness gap, then the proof size $l(n)$ satisfies

$$\log(t\sqrt{g(n)}) \leq l(n) \leq t' \log(n),$$

for constants t, t' and $n > 1$.

In Theorems 9.3.4, 9.3.5 and 9.3.6, the norm used to measure distances was the general trace norm. By making $\epsilon = \delta \leq \frac{k}{\dim(\mathcal{K})^2}$, for a suitable constant k , and using the **SEP** version of the GNN protocol in Theorem 9.3.8, we see that the same results hold for the more restricted Φ^{SEP} disentangler.

9.4 Connection to the de Finetti Theorems

Quantum de Finetti Theorems provide sufficient conditions that limit the maximum correlations that quantum states on n subsystems may exhibit when considering only k ($k \leq n$) of them. They are important tools to limit the entanglement among these subsystems.

Brandão et al. established the connection of a version of the de Finetti Theorem for parallel **LOCC** to $(\epsilon, 0, f(x) = \frac{tx}{2})$ -disentangler in this norm, where t is a constant [20] [18]. We define a generic de Finetti Theorem based on [68], and observe that it leads to a generic disentangler.

Theorem 9.4.1 (Generic de Finetti Theorem). *Let $\rho_{A_1 \dots A_n}$ be a permutation invariant state on $\mathcal{H}_A^{\otimes n}$. Then, for integers $0 \leq k \leq n$, there exists a probability measure μ on density matrices on \mathcal{H}_A , and a function g such that*

$$\left\| \rho_{A_1 \dots A_k} - \int \sigma^{\otimes k} d\mu(\sigma) \right\|_M \leq g(|A|, k, n).$$

This generic theorem implies the existence of the following generic disentangler.

Lemma 9.4.2. *A de Finetti theorem, in the generic form of Theorem 9.4.1, implies the existence of a $(\epsilon, 0, f)$ -disentangler $\Phi_{=}^M$, and where the error ϵ is given by $g(|A|, k, n)$ with function f satisfying $f(x) = nx$.*

Proof. Let $\rho_{A_1 \dots A_n}$ be a state in $\mathcal{H}_A^{\otimes n}$, and let $k = 2$. The disentangler $\Phi_{=}^M$ selects uniformly at random one permutation τ in the symmetric group S_n , and permute the systems $A_1 \dots A_n$ with respect to it. Then, it traces out all of them, except the first two subsystems which we denote by A'_1 and A'_2 . This action can be described as

$$\Phi_{=}^M(\rho_{A_1 \dots A_n}) = \text{Tr}_{\setminus A'_1 A'_2} \left(\sum_{\tau \in S_n} \frac{1}{n!} \tau \rho_{A_1 \dots A_n} \tau^\dagger \right).$$

After a random permutation, the state becomes permutation invariant. Hence, the generic de Finetti Theorem 9.4.1 applies [50]. Let $\rho'_{A'_1 A'_2}$ denote the output state. The de Finetti Theorem guarantees that it is ϵ -close where ϵ is given by

$$\epsilon \leq g(|A|, 2, n).$$

Conversely, any separable state $\sigma = \int \psi \otimes \psi d\mu'(\psi)$, where μ' is a measure on density matrices, can be extended to n subsystems $\sigma^n = \int \psi^{\otimes n} d\mu'(\psi)$. This new state is permutation invariant, and its reduced state in A'_1 and A'_2 is equal to σ . Therefore, the error δ is zero. The input space is $\mathcal{H} = \mathcal{H}_A^{\otimes n}$, and the output space is $\mathcal{K}^{\otimes 2} = \mathcal{H}_a^{\otimes 2}$. Thus $f(x) = nx$. \square

Theorem 9.4.3. *For any constant $p > p_0$, unless there is a $o(2^{\sqrt{n}})$ time quantum algorithm for 3SAT, the following quantum de Finetti Theorem is impossible.*

Let $\rho_{A_1 \dots A_n}$ be a permutation invariant state on $\mathcal{H}_A^{\otimes n}$. Then, for integers $0 \leq k \leq n$ there exists a probability measure μ on density matrices on \mathcal{H}_A such that

$$\left\| \rho_{A_1 \dots A_k} - \int \sigma^{\otimes k} d\mu(\sigma) \right\|_{\mathbf{SEP}} \leq \frac{\text{poly}(k) \text{poly}(|A|)}{n^p}.$$

Proof. We show that if the stated de Finetti theorem is possible, then there is a (ϵ, δ, f) -disentangler $\Phi_{=}^{\mathbf{SEP}}$, with $f(x) = 2^{\frac{x}{c}}$ and $\epsilon, \delta \leq \frac{k'}{\dim(\mathcal{K})^2}$, for some constants c and k' . Combining this result with the extension of Theorem 9.3.6 in the \mathbf{SEP} norm implies that 3SAT has a $o(2^{\sqrt{n}})$ time quantum algorithm.

Let the number of output systems k be 2. Let $x = \log(\dim(\mathcal{H}_A))$ and let $n = \frac{2^{\frac{x}{c}}}{x}$. Now, using the de Finetti Theorem we have:

$$\begin{aligned} \left\| \rho_{A_1 A_2} - \int \sigma^{\otimes 2} d\mu(\sigma) \right\|_{\mathbf{SEP}} &\leq \frac{\text{poly}(2) \text{poly}(|A|)}{n^p} \\ &= O\left(\frac{|A|^a}{2^{\frac{p}{c}x - p \log(x)}}\right) \\ &= O\left(\frac{2^{ax}}{2^{\frac{p}{c}x - p \log(x)}}\right) \\ &= O\left(\frac{1}{2^{\frac{p}{c}x - ax - p \log(x)}}\right), \end{aligned}$$

where a is the maximum degree of $\text{poly}(|A|)$.

For any $p > p_0 = c(a + 2)$, we have the following upper bound

$$O\left(\frac{1}{\dim(\mathcal{H}_A)^{2+\epsilon}}\right).$$

for any $\epsilon > 0$. This is asymptotically smaller than $O(\frac{k'}{\dim(\mathcal{H}_A)^2})$. Note that the random permutation takes time at most $O(\text{poly}(2^x))$ since the disentangler input size is $O(2^{\frac{x}{c}})$. This leads to a disentangler that contradicts our hardness assumption. \square

Note 9.4.4. *Given the hardness assumption of the previous Theorem, even for the restricted \mathbf{SEP} norm the distance error from a separable state in the de Finetti Theorem does not decrease faster than an inverse polynomial in the number of subsystems n . But this dependence is $\frac{1}{n}$ for the more general trace norm version of the de Finetti Theorem [24] [97].*

Note 9.4.5. *The previous Theorem holds for a different constant p_0 , even if the dependence on the dimension subsystem A is polylogarithmic.*

Note 9.4.6. *Any proof that $\text{QMA}(2) \subseteq \text{QMA}$ relying only on a de Finetti result of the same form as Theorem 9.4.1 must have a polylogarithmic dependence on A , since the dependence on n in the best case is an inverse polynomial, assuming the hardness assumption.*

9.5 Discussion

The starting point of our hardness results for disentanglers and the generic quantum de Finetti Theorem is a protocol for 3SAT in $\text{QMA}(2)_{O(\log(n))}$ which has completeness soundness gap of $\Omega(\frac{1}{n^{\text{polylog}(n)}})$. One way to strengthen these hardness results would be to devise protocols with larger gaps. For an (ϵ, δ, f) -disentangler, the errors ϵ and δ are directly related to this gap. Therefore, an interesting research direction is to improve this gap, or otherwise show that it is optimum.

Chapter 10

Area Law

10.1 Introduction

The area law is an important witness that Computer Science can be used as a lens to understand other sciences. In this case, this other science is Physics, more specifically, condensed matter. Let H be the Hamiltonian governing a system \mathcal{S} and let $\lambda_0(H) \leq \lambda_1(H) \leq \dots$ be its eigenvalues. We say that H is gaped if $\lambda_1(H) - \lambda_0(H)$ is a constant greater than zero. The area law in its full generality conjectures that the entanglement across a bipartition (A, \bar{A}) of \mathcal{S} scales according to the surface of A (denoted by $\partial(A)$) rather than its volume, when H is gaped. The entanglement can be measured using a variety of measures which we denote generically by **EM**. This measure can be the entropy of the reduced state on partition A , such as the von Neumann entropy. The area law can be stated in a more rigorous way as follows.

Conjecture 10.1.1 (General Area Law). *Let \mathcal{S} be a system governed by a gaped Hamiltonian H and let (A, \bar{A}) be a bipartition of it. The area law states that*

$$\mathbf{EM}(A, \bar{A}) \leq O(\partial(A)).$$

An example of bipartition in two dimensions (2-D), where the systems lie in a grid, is depicted next.

Understanding the behavior of entanglement may unveil properties of materials. For this reason, the area law has an important application in the designing of new materials. For this thesis, the area law shows the important role of entanglement from a mixed perspective of Computer Science and Physics.

Despite being a conjecture for the 2-D case, the area law actually holds for the 1-D case where the systems lie in a line. The goal of this section is to prove the 1-D area law for a frustration free Hamiltonian H ($\lambda_0(H) = 0$). This result also holds when $\lambda_0(H) \neq 0$,

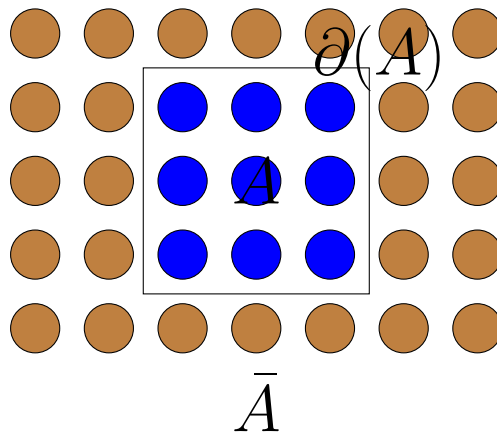


Figure 10.1: Partition A is formed by the systems inside the rectangle. The surface of this region is denoted by $\partial(A)$.

but the simpler frustration free case already requires the main tools without the burden of excessive details. Our proof is based on the results of [8] and [9]. Moreover, it heavily relies on lecture notes of [95]. Despite living in an exponentially large Hilbert space, finding a good approximation for the ground state of 1-D gaped systems can be done efficiently [67].

We establish the notation and assumptions necessary for the 1-D area law. Let n be the number of systems in the line which we consider to be qubits. The Hamiltonian $H = \sum_{i=1}^{n-1} H_i$ is composed of $n-1$ local terms where each one, $0 \leq H_i \leq 1$, is a projection ($H_i^2 = H_i$) acting on qubits i and $i+1$. The ground state $|\psi_0\rangle$ is unique and H is frustration free. Moreover, the spectral gap is a constant $\delta = (\lambda_1(H) - \lambda_0(H)) > 0$.

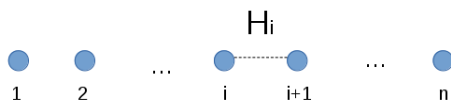


Figure 10.2: 1-D Chain.

10.2 Approximate Ground State Projector

The central tool in our approach for the area law is the Approximate Ground State Projector (AGSP) relative to a certain cut of qubits (A, \bar{A}) . Since we are working in 1-D, a cut can be specified by the pair $(i^*, i^* + 1)$ where i^* indexes a qubit in the chain 10.2. When applied to a quantum state, the AGSP operator leaves the ground state $|\psi_0\rangle$ invariant while shrinking components orthogonal to it. One undesired side-effect of this

operation is that it increases the bond dimension across the associated cut. These two properties of the AGSP are captured by the parameters Δ and B whose precise definition is shown next.

Definition 10.2.1 (AGSP). *Given a Hamiltonian H , an (B, Δ) -AGSP for H with respect to the cut of qubits i^* and $i^* + 1$ is an operator $K \in (\mathbb{C}^2)^{\otimes n}$ such that*

- $K|\psi_0\rangle = |\psi_0\rangle$;
- For all states $|\psi\rangle$ such that $\langle\psi|\psi_0\rangle = 0$, we have $\|K|\psi\rangle\|^2 \leq \Delta\|\psi\|^2$;
- In terms of tensor network representation, the bond dimension of state $K|\psi\rangle$ is at most B times its value for state $|\psi\rangle$ across the cut i^* and $i^* + 1$.

Note that we work with the bond dimension or the Schmidt Rank (SR) parameter, since it is simple to bound and it can latter help bounding the von Neumann entropy (S) across the cut i^* and $i^* + 1$. At each application of an AGSP, we may get closer to the ground state, but we may also increase the entropy. The appropriate trade-off between B and Δ , satisfying $B\Delta \leq \frac{1}{2}$, captures a right balance between the rate of convergence to the ground state and the rate of increase in the entropy that is necessary to establish the area law.

Theorem 10.2.2. *Suppose there exists an (B, Δ) -AGSP such that $B\Delta \leq \frac{1}{2}$. Then $|\psi_0\rangle$ satisfies an area law of the form $S(A)_{|\psi_0\rangle} \leq O(1) \log B$.*

In our 1-D case, the boundary of any bi-partition comprises a constant number of qubits. Thus the area law states that the entanglement is bounded by a constant independent of the number of qubits n . For a constant B and the appropriate AGSP, the previous theorem implies the 1-D area law.

Theorem 10.2.3 (1-D Area Law). *$|\psi_0\rangle$ satisfies an area law of the form $S(A)_{|\psi_0\rangle} \leq O(1)$.*

To prove Theorem 10.2.2, we first show the existence of a product state that has a minimum overlap with the ground state. Then, starting from this state we successively apply the AGSP to get better and better approximations of the ground state while not increasing the bound dimension too much. The next lemma achieves the first goal.

Lemma 10.2.4 (Adapted from [95]). *Suppose there exists an (B, Δ) -AGSP such that $B\Delta \leq \frac{1}{2}$. Fix a partition (A, \bar{A}) of the space on which the Hamiltonian acts. Then there exists a product state $|\phi\rangle = |L_A\rangle \otimes |R_{\bar{A}}\rangle$ such that $|\langle\phi|\psi_0\rangle| = \mu \geq \frac{1}{\sqrt{2B}}$.*

Proof. Let $|\phi\rangle = \mu|\psi_0\rangle + \sqrt{1-\mu^2}|\psi_0^\perp\rangle$ be the state that maximizes the overlap $|\langle\phi|\psi_0\rangle| = \mu$. We apply the AGSP K to it, obtaining the state $|\phi'\rangle = K|\phi\rangle = \mu|\psi_0\rangle + \delta'|\psi_0'^\perp\rangle$ with $|\delta'|^2 \leq \Delta(1-\mu^2) \leq \Delta$. Let $\sum_{i=1}^B \lambda_i |L_i\rangle |R_i\rangle$ be the Schmidt decomposition of $|\phi'\rangle$. Note that the Schmidt rank is at most B since K is an (B, Δ) -AGSP.

We bound the overlap of $|\psi_0\rangle$ and $|\phi'\rangle$ as

$$\begin{aligned} |\langle\psi_0|K|\phi\rangle| &= \left| \sum_{i=1}^B \lambda_i \langle\psi_0|L_i\rangle |R_i\rangle \right| \\ &\leq \sqrt{\sum_{i=1}^B \lambda_i^2} \sqrt{\sum_{i=1}^B (|\langle\psi_0|L_i\rangle |R_i\rangle|)^2} \quad (\text{Cauchy-Schwarz}) \\ &\leq \sqrt{\mu^2 + \Delta} \sqrt{B} \sqrt{\max_i (|\langle\psi_0|L_i\rangle |R_i\rangle|)^2} \\ &= \sqrt{\mu^2 + \Delta} \sqrt{B} \max_i |\langle\psi_0|L_i\rangle |R_i\rangle|. \end{aligned}$$

Let $|L_i\rangle |R_i\rangle$ be the term that maximizes $\max_i |\langle\psi_0|L_i\rangle |R_i\rangle|$. The overlap $|\langle\psi_0|L_i\rangle |R_i\rangle|$ is no greater than the overlap of $|\phi|\psi_0\rangle| = \mu$. The following inequality holds

$$\mu \geq |\langle\psi_0|L_i\rangle |R_i\rangle| \geq \frac{\mu}{\sqrt{B}\sqrt{\mu^2 + \Delta}}.$$

Rearranging the terms results in

$$\sqrt{B}\sqrt{\mu^2 + \Delta} \geq 1.$$

Using the hypothesis that $B\Delta \leq \frac{1}{2}$, we have

$$\begin{aligned} \mu^2 &\geq \frac{1}{B} - \Delta \\ &\geq \frac{1}{B} - \frac{1}{2B} \\ &= \frac{1}{2B}. \end{aligned}$$

This concludes the proof as $\mu \geq \frac{1}{\sqrt{2B}}$. \square

Before we move to the next lemma, we need an auxiliary result known as the Eckart-Young Theorem. This result bounds the maximum possible overlap between states $|\psi\rangle$ and $|\phi\rangle$, where the latter has a predefined Schmidt Rank.

Theorem 10.2.5 (Eckart-Young). *Let $|\psi\rangle \in \mathbb{C}^{d_1} \otimes \mathbb{C}^{d_2}$ be a normalized vector with Schmidt decomposition $|\psi\rangle = \sum \lambda_i |u_i\rangle |v_i\rangle$. Then for any normalized $|\phi\rangle \in \mathbb{C}^{d_1} \otimes \mathbb{C}^{d_2}$ with Schmidt decomposition $|\phi\rangle = \sum_{j=1}^B \mu_j |x_j\rangle |y_j\rangle$, it holds*

$$|\langle \psi | \phi \rangle| \leq \sqrt{\sum_{i=1}^B \lambda_i^2}.$$

Given a product state that has overlap μ with the ground state, the next lemma uses a generic AGSP to bound the entanglement across the associated cut $(i^*, i^* + 1)$.

Lemma 10.2.6. *Suppose there exist an (B, Δ) -AGSP and a product state $|\phi\rangle = |L\rangle_A \otimes |R\rangle_{\bar{A}}$ such that $|\langle \phi | \psi_0 \rangle| = \mu$, then it holds that*

$$S((i^*, i^* + 1))_{|\psi_0\rangle} \leq O(1) \frac{\log \mu}{\log \Delta} \log B.$$

Proof. Denote by K the (B, Δ) -AGSP. We define $|\phi_l\rangle = \frac{K^l |\phi\rangle}{\|K^l |\phi\rangle\|} = \mu |\psi_0\rangle + \sqrt{1 - \mu^2} |\psi_0^\perp\rangle$, where $|\psi_0^\perp\rangle$ is a normalized state orthogonal to the ground state. Note that $|\phi_l\rangle$ is such that

- (i) $SR(|\phi_l\rangle) \leq B^l$;
- (ii) $|\langle \psi_0 | \phi_l \rangle| \geq \frac{\mu}{\sqrt{\mu^2 + \Delta^l(1 - \mu^2)}}$.

Property (i) follows from $SR(|\phi\rangle) = 1$ and the AGSP having bond parameter B . Furthermore, property (ii) is a consequence of the shrinking parameter Δ .

Let $|\psi_0\rangle = \sum \lambda_i |L_i\rangle |R_i\rangle$ be the Schmidt decomposition of the ground state relative to the cut $(i^*, i^* + 1)$. By the Eckart-Young Theorem we have

$$\sum_{i=1}^{B^l} \lambda_i^2 \geq |\langle \psi_0 | \phi_l \rangle|^2 \geq \frac{\mu^2}{\mu^2 + \Delta^l(1 - \mu^2)},$$

or equivalently

$$\sum_{i > B^l} \lambda_i^2 \leq 1 - \frac{\mu^2}{\mu^2 + \Delta^l(1 - \mu^2)} \leq 1 - \frac{\mu^2}{\mu^2 + \Delta^l} \leq \frac{\Delta^l}{\mu^2}.$$

We choose $l_0 = 2 \frac{\log \mu}{\log \Delta} - \frac{\log 2}{\log \Delta}$ such that $\frac{\Delta^{l_0}}{\mu^2} \leq \frac{1}{2}$ and proceed to bound the worst case entropy across the AGSP cut. The first B^{l_0} Schmidt coefficients account for an entropy of at most $l_0 \log B$. For the remaining coefficients, we group them in chunks of size B^{l_0} in intervals $[B^{kl_0} + 1, B^{(k+1)l_0}]$ indexed by k . For each of these intervals, the corresponding entropy can be upper bounded by

$$\frac{\Delta^{kl_0}}{\mu^2} \log B^{(k+1)l_0} = l_0 \frac{1}{2^k} (k+1) \log B.$$

Therefore, the total entropy is

$$\begin{aligned}
S((i^*, i^* + 1)) &\leq l_0 \log B + \sum_{k \geq 1} l_0 \frac{1}{2^k} (k + 1) \log B \\
&\leq l_0 \log B + l_0 \log B \sum_{k \geq 1} \frac{1}{2^k} (k + 1) \\
&\leq O(1) l_0 \log B \\
&\leq O(1) \frac{\log \mu}{\log \Delta} \log B - O(1) \frac{1}{\log \Delta} \log B.
\end{aligned}$$

To simplify the exposition, we have proved a weaker bound that also implies the area law for constant Δ . For a tighter bound, please see [9]. \square

The Detectability Lemma gave us an AGSP with $B = 4$ and $\Delta = 1 - \Omega(\delta)$. Unfortunately, the product $B\Delta \approx 4$ when the gap δ is small does not satisfy our requirement. The proof of the next theorem shows that there is indeed an AGSP with the desired properties.

Theorem 10.2.7. *There is an (B, Δ) -AGSP with $B\Delta \leq \frac{1}{2}$.*

To prove this theorem, we first claim some properties about Chebyshev polynomials and we apply them to the Hamiltonian H to construct the appropriate AGSP.

One important property that we make use of is the subadditivity of the Schmidt Rank, as stated in the following claim. After applying a polynomial to our Hamiltonian $H = \sum_{i=1}^{n-1} H_i$, we end up with a sum of products. The subadditivity implies that the Schmidt Rank of the sum is bounded by the Schmidt Rank of these products.

Claim 10.2.8. $SR(|\psi\rangle + |\phi\rangle) \leq SR(|\psi\rangle) + SR(|\phi\rangle)$.

Using Chebyshev polynomials, it is possible to construct a polynomial that maps 0 to 1 and shrinks the values in a given interval.

Claim 10.2.9. *For any integer l , there exists a real polynomial P_l such that*

- $P_l(0) = 1$;
- For all $x \in [\delta, \lambda^*]$, $|P_l(x)| \leq \sqrt{\Delta} = 2e^{-2l\sqrt{\frac{\delta}{\lambda^*}}}$.

A sketch of a possible Chebyshev polynomial is shown next.

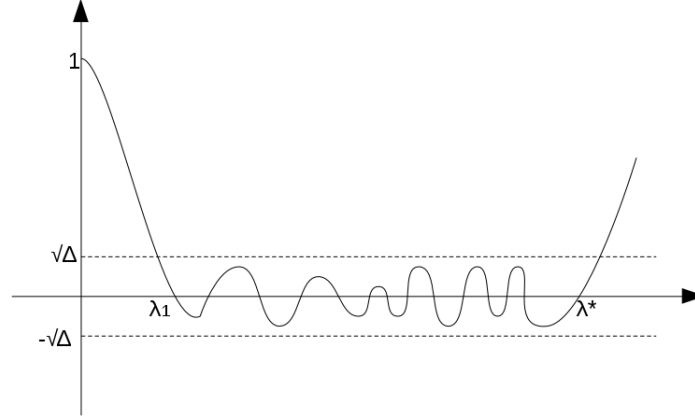


Figure 10.3: Sketch of a Chebyshev Polynomial.

If we apply this polynomial to our Hamiltonian H that is Hermitian, it is equivalent to apply the polynomial to its eigenvalues, in which case we have an operator $K_l = P_l(H)$ satisfying

- (i) $K_l|\psi_0\rangle = |\psi_0\rangle$ since $P_l(0) = 1$;
- (ii) if $|\langle\psi|\psi_0\rangle| = 0$, it has eigenvalue between $[\delta, \lambda^*]$ where δ is the gap and λ^* is the operator norm of H and $\|K_l|\psi\rangle\| \leq \Delta$;
- (iii) $SR(K_l) \leq (l+1)n^l 4^l$.

It is clear that $P_l(H)$ is an $(B = (l+1)n^l 4^l, \Delta = 4e^{-4l}\sqrt{\frac{\delta}{\lambda^*}})$ -AGSP. Note that the bound $SR(K_l) \leq (l+1)n^l 4^l$ was calculated in a very naive way, the multivariate polynomial of degree l , $P_l(H_1 + \dots + H_{n-1})$, has at most $(l+1)n^l$ terms where each has a Schmidt Rank bounded by 4^l . Then, using the subadditivity of the Schmidt Rank the bound follows. The dependence on the number of qubits is undesired since the 1-D area law must bound entanglement by a constant. Fortunately, there is a clever way to bound B that results in $B\Delta \leq \frac{1}{2}$ as we explore next.

We first replace H by

$$\tilde{H} = H_L + H_{i-\frac{u}{2}} + \dots + H_i + \dots + H_{i+\frac{u}{2}-1} + H_R,$$

where $H_L = \text{trunc}(H_1 + \dots + H_{i-\frac{u}{2}-1}, t)$ and $H_R = \text{trunc}(H_{i+\frac{u}{2}} + \dots + H_{n-1}, t)$. The operation $\text{trunc}(X, t)$, when applied to an Hermitian operator X , truncates its eigenvalues above t to t . Therefore, it is clear that the operator norm of \tilde{H} is $\lambda^* \leq u + 2t$. Moreover, this process has not changed the ground. A non-trivial observation is that $\tilde{\delta} = \lambda_1(\tilde{H}) \geq \Omega(\lambda_1(H) = \delta)$ for $t = O(\frac{1}{\delta})$. In other words, the new spectral gap is related to the original gap by a constant.

Lemma 10.2.10. *If $\sqrt{l} \geq u$, then $K = P_l(\tilde{H})$ is a (B, Δ) -AGSP with $\Delta \leq 4e^{-4l\sqrt{\frac{\tilde{\delta}}{\lambda^*}}}$ and $B \leq (2l)^{O(\frac{l}{u})}$*

Let's first verify if this lemma can actually lead to $B\Delta \leq \frac{1}{2}$. For a truncated bound $t = \frac{c_1}{\delta}$, we have a spectral gap $\tilde{\delta} = c_2\delta$ for \tilde{H} , as noted before. For some constant c_3 , multiplying B and Δ results in

$$\begin{aligned} B\Delta &\leq 4e^{-4l\sqrt{\frac{\tilde{\delta}}{\lambda^*}}}(2l)^{c_3\frac{l}{u}} \\ &\leq 4e^{-4l\sqrt{\frac{\tilde{\delta}}{\lambda^*}} + c_3\frac{l}{u}\ln 2l}. \end{aligned}$$

To achieve $B\Delta \leq \frac{1}{2}$, the following must hold

$$\begin{aligned} -\ln 8 &\geq -4l\sqrt{\frac{\tilde{\delta}}{\lambda^*}} + c_3\frac{l}{u}\ln 2l \\ &= -4l\sqrt{\frac{c_2\delta}{2t+u}} + c_3\frac{l}{u}\ln 2l \\ &= -4l\sqrt{\frac{c_2\delta}{2\frac{c_1}{\delta}+u}} + c_3\frac{l}{u}\ln 2l. \end{aligned}$$

Making $u = \sqrt{l}$, we have

$$-4l\sqrt{\frac{c_2\delta}{2\frac{c_1}{\delta}+\sqrt{l}}} + c_3\sqrt{l}\ln 2l \leq -\ln 8.$$

Note that the term $l\sqrt{\frac{c_2\delta}{2\frac{c_1}{\delta}+\sqrt{l}}}$ is asymptotically bigger than $c_3\sqrt{l}\ln 2l$. Therefore, there is an l sufficiently large that satisfies this inequality.

Knowing that we are going in the right direction, we proceed to prove the lemma.

Proof. As noted before, we need a clever way to upper bound the Schmidt Rank. Let $(i^*, i^* + 1)$ be the cut of the AGSP. The polynomial $P_l(\tilde{H})$ is a linear combination of $l + 1$ powers of \tilde{H} . We bound each of these terms by the one with the greatest Schmidt Rank namely \tilde{H}^l . It is possible to introduce commuting variables Z_0, \dots, Z_{u+1} , and write, as a generating function of them based on \tilde{H}^l ,

$$(H_L Z_0 + \dots + H_R Z_{u+1})^l = \sum_{a_0 + \dots + a_{u+1} = l} f_{a_0, \dots, a_{u+1}} Z_0^{a_0} \dots Z_{u+1}^{a_{u+1}}.$$

The role of each of these variables is to count the number of occurrences of H_L, \dots, H_R . An observation that is crucial to improve our previous naive analysis is the existence of

an $i \in \{1, \dots, u\}$ such that $a_i \leq \frac{l}{u}$, since $a_0 + \dots + a_{u+1} = l$. This means that for each multi-index (a_0, \dots, a_{u+1}) there is a cut $(i, i+1)$ whose corresponding Hamiltonian is not applied too much, thus being a good candidate for bounding the blow up in the Schmidt Rank. From this new perspective, \tilde{H} can be rewritten as $\tilde{H}^l = \sum_{i=1}^u \sum_{k=0}^{\frac{l}{u}} Q_{i,k}$. Each term $Q_{i,k}$ is a sum of some operators $f_{a_0, \dots, a_{u+1}}$ where $a_i = k$ and $\sum_{j \neq i} a_j = l - k$.

We proceed by bounding the Schmidt Rank of $Q_{i,k}$. Firstly, we introduce the following generating function using the same variables introduced before, except for Z_i ,

$$P_{i,k}(Z) = \sum_{a_0 + \dots + a_{u+1} = l, a_i = k} f_{a_0, \dots, a_{u+1}} \Pi_{j \neq i} Z_j^{a_j}.$$

The sum above contains $t = \binom{l-k+u}{u}$ terms. By considering the set of operators $\{f_{a_0, \dots, a_{u+1}} | a_0 + \dots + a_{u+1} = l \text{ and } a_i = k\}$ a basis for a linear space in the field \mathbb{C} , it is possible to describe $Q_{i,k}$ as a linear combination of at most t terms $P_{i,k}(Z)$ in which we assign a value $Z \in \mathbb{C}^{u+1}$ for each of them. Now, we need to bound the Schmidt Rank of $P_{i,k}(Z)$ for a fixed Z . For this, we explore the expression $(A + H_i + B)^l$, where $A = \sum_{j < i} H_j Z_j$ and $B = \sum_{j > i} H_j Z_j$. Observe that A and B commute since they act on different qubits. Rearranging \tilde{H}^l by grouping terms that contains k occurrences of H_i , we have

$$(A + H_i + B)^l = \sum_{k=0}^l \sum_{(a_0+b_0)+\dots+(a_k+b_k)=l-k} A^{a_0} B^{b_0} H_i \dots H_i A^{a_k} B^{b_k}.$$

For each k , the above sum has a total of $\binom{l+k}{2k+1}$ terms whose Schmidt Rank can be bounded by 2^{2k} . All the reasoning was done for a specific favorable cut $(i, i+1)$ that does not necessarily correspond to the cut $(i^*, i^* + 1)$ of our AGSP. We pay a multiplicative cost of at most $2^{2|i-i^*|} \leq 2^u$ in the bond dimension of our original AGSP cut.

Combining all the elements, and using the fact that $k \leq \frac{l}{u}$ and $u \leq \sqrt{l}$, we are ready to bound $SR(Q_{i,k})$ as

$$SR(Q_{i,k}) \leq \binom{l-k+u}{u} \binom{l+k}{2k+1} 2^{2k+u} \leq l^{O(u)} l^{O(\frac{l}{u})} 2^{2\frac{l}{u}+u} \leq (2l)^{O(\frac{l}{u})}.$$

The summation of $\tilde{H}^l = \sum_{i=1}^u \sum_{k=0}^{\frac{l}{u}} Q_{i,k}$ over i and k can be absorbed in the asymptotic notation used in the exponent.

The bound on Δ follows from the previous claim about Chebyshev polynomials. \square

Chapter 11

On the Languages Recognized by 2QCFA

11.1 Introduction

In 1984, Feynman proposed the quantum computational model, in which quantum physics could be simulated efficiently (in polynomial time) [36]. Just a decade latter, Shor [91] was able to create a polynomial time factoring algorithm that can possibly establish an exponential separation regarding the classical computational model. Along with Grover's database search algorithm with a quadratic speedup [47], quantum computing became an important research field. The study of simpler computational models, such as quantum automata, reveals fundamental differences between classical and quantum computation. In this context, the Two-way quantum classical finite state automaton (2QCFA) [7] illustrates what is possible to compute using fixed quantum and classical memory.

There are some variations of quantum automata models. The One-way quantum finite automaton (1QFA) [66] [72] recognizes a proper subset of regular languages. Another variation, Two-way finite automaton (2QFA) [66], recognizes all regular languages, and also some non-regular languages. Nonetheless, a 2QFA uses memory that is proportional to the logarithm of the input word length ; therefore, its memory is not finite. The 2QCFA combines a classical automaton with actual fixed dimensional quantum memory. For this reason, it recognizes all regular languages, being more powerful than the 1QFA model. In addition, it also recognizes context-free languages (CFL), and even context-sensitive languages (CSL).

There is still no characterization of languages that a 2QCFA can recognize. As a result, it is not known the exact relationship of the set of languages recognized by this model, and the classical Chomsky hierarchy. This chapter is mainly a survey of known 2QCFA languages, and closure properties. Additionally, we present a new result showing

that the language $L_{>} = \{a^i b^j | i, j \in \mathbb{N} \text{ and } i > j\}$ can be recognized by a 2QCFA.

This article is structured as follows. Section 11.2 contains the formal definition of the 2QCFA model, and operational error types. Section 11.3 has a list of languages recognized by this model with their complete algorithms; it also encloses the $L_{>}$ recognition proof. Section 11.4 brings a list of closure properties. Finally, the conclusion, section 11.5, contains future research ideas.

11.2 General Definitions

In this section, we define the 2QCFA model, and the their types of language recognition according to the error.

From [7], the 2QCFA model is basically a classical automaton with a constant size quantum memory. Formally, it is defined by a 9-tuple, $M = (Q, S, \Sigma, \Theta, \delta, q_0, s_0, S_{acc}, S_{rej})$, in which:

- Q and S : set of quantum, and classical states, respectively;
- Σ : input alphabet;
- Θ and δ : quantum, and classical evolution functions, respectively;
- q_0 and s_0 : quantum, and classical initial states, respectively;
- S_{acc} and S_{rej} : set of classical accepting and rejecting states, respectively. We assume $S_{acc} \cap S_{rej} = \emptyset$.

A generic input word x is placed in the input tape with two special delimiters, \dagger and $\$$, to mark the beginning, and end of the input, respectively. These two symbols are not part of the input alphabet; the tape alphabet, Γ , is defined as $\Gamma = \{\dagger, \$\} \cup \Sigma$. Tape position 0 has the symbol \dagger . For $i \in [1, |x|]$, tape position i has symbol, x_i , of x . Finally, position $|x| + 1$ has the symbol $\$$. The automaton is not allowed to move the tape head, neither from the left of \dagger , nor to the right of $\$$.

Let $S_{halt} = S_{acc} \cup S_{rej}$ be the set of halting states, and $S_{non} = S \setminus S_{halt}$ be the set of non halting states. Moreover, let s , q , and σ be, respectively, the current classical state, the current quantum state, and symbol under the tape head.

The computation of a 2QCFA starts at states q_0 and s_0 . At each iteration, the automaton uses s and σ to determine the quantum evolution $\Theta(s, \sigma)$, which can be a unitary transformation, or a measurement. Formally, Θ is a mapping $\Theta : S_{non} \times \Gamma \rightarrow \mathcal{U}(\mathcal{H}(Q)) \cup \mathcal{M}(\mathcal{H}(Q))$. The set $\mathcal{U}(\mathcal{H}(Q))$ contains unitary transformations in the fixed dimension Hilbert space whose base states are in Q . Analogously, the set $\mathcal{M}(\mathcal{H}(Q))$

contains projective measurements in the same space and base states. Next, the classical evolution, δ , is applied, but its form depends on the type of quantum evolution previously performed. If it was a measurement, δ is the mapping $\delta : S_{non} \times \Gamma \times R \rightarrow S \times D$, in which R is the set of possible measurement results, and $D = \{-1, 0, +1\}$ is the set of head directions for the tape. These directions left, none, and right are mapped respectively to -1 , 0 , and $+1$. Otherwise, if the last quantum operation was a unitary transformation, δ is the usual classical transition function of the form: $\delta : S_{non} \times \Gamma \rightarrow S \times D$. The computation continues until a halting state is reached.

The probabilistic nature of quantum measurements makes the 2QCFA a probabilistic model. For a given word x , there is a probability p_{acc} of x being accepted, and a probability p_{rej} of it being rejected. The automaton may not halt; therefore, $p_{acc} + p_{rej}$ may be smaller than 1.

11.2.1 Language Recognition

Due to the probabilistic nature of the 2QCFA model, there are three types of language recognition errors.

Definition 11.2.1 (Zero error). *We say that a 2QCFA, M , that recognizes a language, L , with zero error if $x \in L$, then $P[M \text{ accepts } x] = 1$; otherwise $P[M \text{ rejects } x] = 1$.*

Definition 11.2.2 (One-sided error). *We say that a 2QCFA, M , that recognizes a language, L , with one-sided error ϵ if $x \in L$, then $P[M \text{ accepts } x] = 1$; otherwise $P[M \text{ rejects } x] \geq 1 - \epsilon$.*

Note that in our definition, the one-sided error term is used to designate only the case in which the acceptance probability is one if the word is in the language. This decision may seem arbitrary, but it is the one commonly encountered in the literature.

Definition 11.2.3 (Two-sided error). *We say that a 2QCFA, M , that recognizes a language, L , with error ϵ if $x \in L$, then $P[M \text{ accepts } x] \geq 1 - \epsilon$; otherwise $P[M \text{ rejects } x] \geq 1 - \epsilon$.*

11.3 Languages

In this section, we show important languages recognized by the 2QCFA model. It is worth to note that the ideas developed for recognizing languages $L_{=}$ and L_{pal} are employed in the recognition of several other languages; therefore, they form the currently theoretical foundation of the 2QCFA theory.

11.3.1 Language $L_=$

The language $L_=$ is to be defined next. Moreover, it is classified according to the Chomsky hierarchy and according to the error a 2QCFA makes when recognizing it.

Definition 11.3.1.

$$L_ = \{a^n b^n | n \in \mathbb{N}\}.$$

Language Classification: Context-Free. **Error Type:** One-sided.

The recognition of $L_ =$, as in Definition 11.3.1, was proved in the same article which introduced the 2QCFA [7] model. The words in this language may have an arbitrary length, forcing the automaton to keep some sort of counting that depends on the number of ‘a’s and ‘b’s. Since $L_ =$ is a context-free language, but not a regular language, it is impossible to count using only a set of finite classical states. However, the quantum part of the 2QCFA, even though having a fixed dimensional quantum memory, is able to keep the count. To this end, it suffices to use just one qubit.

Let $|q_0\rangle$ and $|q_1\rangle$ be the single qubit base states. The computation starts with $q = |q_0\rangle$. For each input symbol ‘a’, a rotation by an irrational number ($\sqrt{2}\pi$) is applied clockwise. For each ‘b’, a rotation by the same angle is applied counter-clockwise. If the number of ‘a’s and ‘b’s is the same, the total rotation results in zero degrees. Otherwise, since the rotation angle at each step is irrational, there will be a total rotation that is different from zero. At the end of the input, the qubit is measured.

For a generic word x , if $x \in L_ =$, there is a 1 probability of measuring $|q_0\rangle$. If $x \notin L_ =$, there is a non zero probability of measuring $|q_1\rangle$, but there is also a non zero probability of measuring $|q_0\rangle$. The probability of reading $|q_1\rangle$ is at least $\frac{1}{2l^2}$, where l is the length of x . For this reason, when we read $|q_0\rangle$, we simulate a probability inferior to $\frac{1}{2l^2}$, and only then we accept it.

An algorithmic definition of the 2QCFA $M_{L_ =}$ is given in 25.

For a string $a^n b^{n'}$ with $n \neq n'$, the properties of the rotation by an angle $\sqrt{2}\pi$ will guarantee a non zero probability of measuring state $|q_1\rangle$. Therefore, at each step of the loop there is a certain probability that the input is rejected. The next lemma formalizes this statement.

Lemma 11.3.2 (Adapted from [7]). *If the input is $x = a^n b^{n'}$, and $n \neq n'$, then each step of the loop in the 2QCFA for $L_ =$ rejects with probability at least $\frac{1}{2}(n - n')^2$.*

Proof. We start the execution of the automaton with the quantum state set to $|q_0\rangle$. After $n + n'$ rotations by an angle of $\sqrt{2}\pi$, we have a net rotation of $\sqrt{2}(n - n')$, and the quantum state is the superposition


```

1  Let  $x$  be the input word
2  Let  $l$  be the length of  $x$ 
3  Let  $U_\alpha$  be a rotation by an angle  $\alpha$ 
4  // check classically
5  if  $x \notin a^*b^*$  then
6  |   Reject ;
7  end
8  while true do
9  |   // within  $\dagger$  to  $\$$ 
10 |   for each symbol  $\sigma$  do
11 |       if  $\sigma = a$  then
12 |           |    $q = U_{\sqrt{2}\pi}q$  ;
13 |       end
14 |       else
15 |           |    $q = U_{-\sqrt{2}\pi}q$  ;
16 |       end
17 |   end
18 |   Measure the quantum state ;
19 |   if  $q = |q_1\rangle$  then
20 |       |   Reject ;
21 |   end
22 |   else if simulate probability  $\frac{1}{2^k l^2}$  then
23 |       |   Accept ;
24 |   end
25 end

```

Algorithm 13: $L = 2\text{QCFA}$

$$\cos(\sqrt{2}(n - n')\pi)|q_0\rangle + \sin(\sqrt{2}(n - n')\pi)|q_1\rangle.$$

The probability of measuring $|q_1\rangle$ is $\sin^2(\sqrt{2}(n - n')\pi)$ which is an even function. We need to determine how much $\sqrt{2}(n - n')$ differs from its closest integer k . Since the probability function is even, we can assume without loss of generality that $k < \sqrt{2}(n - n')$. Squaring both sides of this inequality, we have $k^2 < 2(n - n')^2$. It also holds that $k^2 \leq 2(n - n')^2 - 1$, because k^2 is integer and $2(n - n')^2 - 1$ is the closest integer smaller than $2(n - n')^2$. Moreover, we also have

$$\begin{aligned} (\sqrt{2}(n - n') - \sqrt{2(n - n')^2 - 1})(\sqrt{2}(n - n') + \sqrt{2(n - n')^2 - 1}) \\ = 2(n - n')^2 - 2(n - n')^2 + 1 = 1. \end{aligned}$$

Using these facts, we are ready to compute a lower bound for the difference $\sqrt{2}(n - n') - k$ as

$$\begin{aligned} \sqrt{2}(n - n') - k &\geq \sqrt{2}(n - n') - \sqrt{2(n - n')^2 - 1} \\ &= \frac{1}{\sqrt{2}(n - n') + \sqrt{2(n - n')^2 - 1}} \\ &> \frac{1}{2\sqrt{2}(n - n')}. \end{aligned}$$

Since k is the closest integer to $\sqrt{2}(n - n')$, we have that $|\sqrt{2}(n - n') - k| \in [0, \frac{1}{2}]$. The bound on $\sin(\pi x)$ is

$$\sin(\pi x) \geq 2x,$$

for x in this interval holds, because $\sin(\pi x)$ is concave and the inequality holds for both extremes 0 and $\frac{1}{2}$.

Combining the bounds, we can conclude the proof as

$$\begin{aligned} \sin^2(\sqrt{2}(n - n')\pi) &= \sin^2(\sqrt{2}(n - n') - k)\pi \\ &\geq 4(\sqrt{2}(n - n') - k)^2 \\ &\geq 4\left(\frac{1}{2\sqrt{2}(n - n')}\right)^2 \\ &= \frac{1}{2(n - n')^2}. \end{aligned}$$

□

The simulation of probability $\frac{1}{2^{kl^2}}$ can be done by tossing k coins and conducting two random walks on the tape. Assuming the input is in the form $x = aa^*bb^* = a^n b^{n'}$. The

random walks are the same. They start at the first symbol a and continue until the symbol \dagger or $\$$ is reached. At each step, with equal probability, it moves to left or to the right. It is a known result that this random walk has a probability of $\frac{1}{n+n'+1}$ [7].

Theorem 11.3.3. *The 2QCFA $M_{L=}$ recognizes $L=$ with one-sided error ϵ .*

Proof. We prove Theorem 11.3.3 by a case analysis. If $x \in L$, the probability of measuring $|q_1\rangle$ is zero, so $M_{L=}$ will eventually accept x . As a result, x is accepted with probability 1.

If $x \notin L$, the probability of accepting at each iteration is $p_{acc} \leq \frac{1}{2^{kl^2}}$, where k is a constant that controls the error ϵ . The probability of rejecting x is, at least, $p_{rej} \geq \frac{1}{2l^2}$. The total probability of rejecting, P_{rej} , is

$$P_{rej} \geq \sum_{i \geq 0} (1 - p_{acc} - p_{rej})^i p_{rej}.$$

Using the sum of a geometric progression results in

$$P_{rej} \geq \frac{p_{rej}}{p_{acc} + p_{rej}} = \frac{1}{1 + \frac{p_{acc}}{p_{rej}}}.$$

Making $\epsilon = \frac{p_{acc}}{p_{rej}}$, we have

$$P_{rej} \geq \frac{1}{1 + \epsilon} = \frac{1(1 - \epsilon)}{(1 + \epsilon)(1 - \epsilon)} \geq 1 - \epsilon.$$

Since $\epsilon = \frac{1}{2^k}$, the constant k is used to control the error ϵ .

□

11.3.2 Language L_{pal}

The language of the palindromes over a binary alphabet is defined next.

Definition 11.3.4.

$$L_{pal} = \{x | \Sigma = \{a, b\} \text{ and } x = x^r\}.$$

Language Classification: Context-Free. **Error Type:** One-sided.

The palindrome language L_{pal} , defined in 11.3.4, can be recognized by a 2QCFA. Since the 2QCFA is also a 2-way probabilistic finite automaton (2PFA), and no 2PFA can recognize L_{pal} with bounded error, we conclude that the 2QCFA model is strictly more

powerful than the 2PFA model. As was the case with the language $L_=$, the recognition of L_{pal} was also shown when the 2QCFA model was first studied [7].

The input alphabet consists of two symbols: $\Sigma = \{a, b\}$. For each symbol $\sigma \in \Sigma$, a matrix U_σ is defined as follows:

$$U_a = \frac{1}{5} \begin{pmatrix} 4 & 3 & 0 \\ -3 & 4 & 0 \\ 0 & 0 & 5 \end{pmatrix}, U_b = \frac{1}{5} \begin{pmatrix} 4 & 0 & 3 \\ 0 & 5 & 0 \\ -3 & 0 & 4 \end{pmatrix} \quad (11.1)$$

The idea is to process the input word from \dagger to $\$$ applying for each symbol its corresponding unitary transformation. Afterwards, the tape head is returned to the first input symbol, and for each symbol the inverse operation is applied. This automaton works with three quantum base states: $|q_0\rangle$, $|q_1\rangle$, and $|q_2\rangle$. Its operation is executed in the sphere of unity radius in \Re^3 , making it possible to map this ternary bit to a regular quantum bit using the Bloch Sphere.

For a given word $x = \sigma_1\sigma_2 \cdots \sigma_l$, where l is its length, the first sequence of operations results in

$$U_{\sigma_l} \cdots U_{\sigma_2} U_{\sigma_1} |q_0\rangle.$$

After applying the second sequence of operations, the quantum state becomes

$$U_{\sigma_l}^{-1} \cdots U_{\sigma_2}^{-1} U_{\sigma_1}^{-1} U_{\sigma_l} \cdots U_{\sigma_2} U_{\sigma_1} |q_0\rangle.$$

If x is a palindrome, then $\sigma_1 = \sigma_l$, $\sigma_2 = \sigma_{l-1}$, and so on. It is clear that the sequence of unitary operation results in the identity, and the final quantum state will be $|q_0\rangle$. On the other hand, if $x \notin L_{pal}$, there will be at least one pair of matrix multiplication, one in each sequence of operations, which does not correspond to U_σ and U_σ^{-1} , for some σ . Given the appropriate choice of U_a and U_b , this will result in a probability at least $\frac{1}{25^l}$ of measuring a state different from $|q_0\rangle$.

As was the case with $L_=$, after measuring state $|q_0\rangle$, it is not possible to distinguish whether the input word belongs to L_{pal} , or not. For this reason, a probability inferior to $\frac{1}{25^l}$ is simulated.

The algorithmic description of the 2QCFA $M_{L_{pal}}$ is given in 23.

We need to establish some notation before proving the acceptance probability of the 2QCFA for L_{pal} . Based on the unitaries U_a and U_b , we defined two matrices whose entries are all in \mathbb{Z} as

$$A = 5U_a, B = 5U_b. \quad (11.2)$$

We define a set K as

$$K = \{u \in \mathbb{Z}^3 : u[1] \not\equiv 0 \pmod{5}, f(u) \not\equiv 0 \pmod{5}, \text{ and } u[2]u[3] \equiv 0 \pmod{5}\},$$

```

1 Let  $x$  be the input word
2 Let  $l$  be the length of  $x$ 
3 Let  $U_\sigma$  be the unitary operation for  $\sigma \in \Sigma$ 
4 while true do
5   | Set quantum state to  $|q_0\rangle$  ;
6   | // from  $\dagger$  to  $\$$  (within)
7   | for each  $\sigma$  do
8   |   |  $q = U_\sigma q$  ;
9   | end
10  | // from  $\dagger$  to  $\$$  (within)
11  | for each  $\sigma$  do
12  |   |  $q = U_\sigma^{-1} q$  ;
13  | end
14  | Measure quantum state;
15  | if  $q \neq |q_0\rangle$  then
16  |   | Reject ;
17  | end
18  | else
19  |   | if Simulate probability  $\frac{1}{2^{kl}}$  then
20  |     | Accept ;
21  |   | end
22  | end
23 end

```

Algorithm 14: L_{pal} 2QCFA

where $f : \mathbb{Z}^3 \rightarrow \mathbb{Z}$ is

$$f(u) = 4u[1] + 3u[2] + 3u[3].$$

Vectors in this set are stabilized by multiplications by A and B . That is $Au \in K$ and $Bu \in K$, if $u \in K$. The next lemma formalizes this fact.

Lemma 11.3.5 (From [7]). *If $u \in K$, then $Au \in K$ and $Bu \in K$.*

Proof. Let $u = (a, b, c)^T$. Let $v = Au$ be

$$v = Au = (4a + 3b, -3a + 4b, 5c).$$

We have that v satisfies $v[2]v[3] \equiv 0 \pmod{5}$ which is a necessary property to be in K . Next, we need to analyze two cases depending on which component of $u[2]$ or $u[3]$ is a multiple of 5.

Firstly, suppose that $b \equiv 0 \pmod{5}$. In this case $(Au)[0]$ is

$$4a + 3b \equiv 4a \pmod{5} \not\equiv 0 \pmod{5}.$$

We also compute $f(v)$ as

$$f(v) = 4(4a + 3b) + 3(-3a + 4b) + 3(5c) \equiv 2a \pmod{5} \not\equiv 0 \pmod{5}.$$

With these three properties, we conclude that v belong to K .

Next, suppose that $c \equiv 0 \pmod{5}$. The component $(Au)[0]$ becomes

$$\begin{aligned} 4a + 3b &\equiv 4a + 3b + 3c \pmod{5} \\ &\equiv f(u) \pmod{5} \not\equiv 0 \pmod{5}. \end{aligned}$$

Furthermore, $f(v)$ is

$$\begin{aligned} f(v) &= 4(4a + 3b) + 3(-3a + 4b) + 3(5c) \equiv 2a + 4b \pmod{5} \\ &\equiv 3(4a + 3b + c) \pmod{5} \\ &\equiv 3f(u) \pmod{5} \not\equiv 0 \pmod{5}. \end{aligned}$$

Therefore, we covered all cases, showing that $Au \in K$. Analogously, for $w = Bu$ we have

$$w = Bu = (4a + 3c, 5b, -3a + 4c).$$

The definition of K is unchanged if we swap the second and third coordinates of a vector. For this reason, we also have that $w \in K$. \square

Lemma 11.3.6 (From [7]). *Let $u \in \mathbb{Z}^3$ satisfy $u = Av = Bw$ for v and w in \mathbb{Z}^3 . Then $u \notin K$.*

Proof. Let $u = Av = Bw$ for $u, v, w \in \mathbb{Z}^3$, so that $A^{-1}u, B^{-1}u \in \mathbb{Z}^3$. Note that $(B^{-1}u)[2] = \frac{5u[2]}{25} \in \mathbb{Z}$, and so $u[2] \equiv 0 \pmod{5}$. Moreover, $(A^{-1}u)[1] = \frac{4u[1]-3u[2]}{25} \in \mathbb{Z}$, and so $4u[1] - 3u[2] \equiv 0 \pmod{25}$. The term $3u[2]$ has 0 or 5 as its least significant digit, given that $u[2] \equiv 0 \pmod{5}$. The least significant digit of any number congruent to 0 modulo 25 is also 0 or 5. Therefore, the term $4u[1]$ must also have the least significant digit as 0 or 5. This implies that $u[1] \equiv 0 \pmod{5}$, proving that $u \notin K$. \square

Lemma 11.3.7 (From [7]). *Let*

$$u = 25^{-n}Y_1^{-1} \cdots Y_n^{-1}X_n \cdots X_1(1, 0, 0)^T,$$

where $X_j, Y_j \in \{A, B\}$. If $X_j = Y_j$ for $1 \leq j \leq n$, then $u[2]^2 + u[3]^2 = 0$. Otherwise, $u[2]^2 + u[3]^2 > 25^{-n}$.

Proof. It is clear that $u = (1, 0, 0)^T$ when $X_j = Y_j$ for $1 \leq j \leq n$, because $Y_j^{-1}X_j$ is just the identity. In this case, $u[2]^2 + u[3]^2 = 0$.

Suppose there is a j such that $X_j \neq Y_j$. We fix k to be the largest index for which $X_k \neq Y_k$. Without loss of generality, assume that $X_k = A$ and $Y_k = B$. Let $v = X_{k-1} \cdots X_1(1, 0, 0)^T$ and $w = Y_{k-1} \cdots Y_1(1, 0, 0)^T$. If $Av = Bw = u$, then by Lemma 11.3.6 we have $u \notin Z$. However, by Lemma 11.3.5 Av and Bw are in Z , contradicting the assumption $Av = Bw$. In other words, we have

$$X_k \cdots X_1(1, 0, 0)^T \neq Y_k \cdots Y_1(1, 0, 0)^T.$$

Since k is the index of the largest disagreement, it holds that $X_j = Y_j$ for $j > k$. For this reason, it also holds that

$$X_n \cdots X_1(1, 0, 0)^T \neq Y_n \cdots Y_1(1, 0, 0)^T.$$

Stated in an equivalent way, we have

$$u = 25^{-n}Y_n^{-1} \cdots Y_1^{-1}X_n \cdots X_1(1, 0, 0)^T \neq (1, 0, 0)^T.$$

Note that the vector $(-1, 0, 0)^T$ also belongs to K , and a analogous reasoning results in

$$u = 25^{-n}Y_n^{-1} \cdots Y_1^{-1}X_n \cdots X_1(1, 0, 0)^T \neq (-1, 0, 0)^T.$$

Since the vector $(1, 0, 0)^T$ was multiplied by unitaries and $u \neq (\pm 1, 0, 0)^T$, we have $|u[1]| < 1$. The vector $25^n u$ has integers components implying that $|u[1]| \leq 1 - 25^{-n}$. Consequently, we can lower bound $u[2]^2 + u[3]^2$ as

$$u[2]^2 + u[3]^2 = 1 - u[1]^2 \geq 1 - (1 - 25^{-n})^2 > 25^{-n}.$$

\square

Theorem 11.3.8. *The 2QCFA $M_{L_{pal}}$ in 23 recognizes L_{pal} with one-sided error ϵ .*

Proof. The proof of 11.3.8 is analogous to that of 11.3.3 for $L_{=}$. Clearly, for a word $x \in L_{pal}$, $\epsilon = 0$. For $x \notin L_{pal}$, the error ϵ of $M_{L_{pal}}$ can be calculated in the same way as

$$\epsilon = \frac{p_{acc}}{p_{rej}},$$

where p_{acc} and p_{rej} are the accepting and rejecting probabilities at each iteration. This results in ϵ as

$$\epsilon = \frac{25^l}{2^{kl}}.$$

Once again, the constant k controls the error. □

11.3.3 Language L_m

Definition 11.3.9.

$$L_m = \{xycy \mid \Sigma = \{a, b, c\} \text{ and } x, y \in \{a, b\}^* \text{ and } |x| = |y|\}.$$

Language Classification: Context-Free. **Error Type:** One-sided.

The language L_m [81] is a simplification of the language L_{middle} which is not known to be recognized by a 2QCFA [7]. It has a special middle symbol ‘ c ’ that is not contained, neither in x , nor in y . With it, the problem of recognizing L_m becomes very similar to the problem of recognizing $L_{=}$.

The algorithmic description of the 2QCFA M_{L_m} is given in 24. A rotation by $\sqrt{2}\pi$ is applied for each symbol before ‘ c ’, and a rotation by $-\sqrt{2}\pi$ is applied for each one after ‘ c ’. If ‘ c ’ is in the middle, then $|x| = |y|$, and the total rotation is zero. Otherwise, $|x| \neq |y|$, and there will be a non zero probability of detecting this unbalanced case.

The algorithmic description of M_{L_m} for language L_m is given in 24.

Theorem 11.3.10. *The 2QCFA M_{L_m} recognizes L_m with one-sided error ϵ .*

Proof. Analogous to the proof of 11.3.3 for $L_{=}$. □

11.3.4 Language L_p

The language L_p is a natural extension of $L_{=}$, in which the number of a ’s is p times the number of b ’s.

Definition 11.3.11.

$$L_p = \{a^{pi}b^i \mid p \in N \text{ and } i \in N\}.$$


```

1 Let  $x$  be the input word
2 Let  $l$  be the length of  $x$ 
3 Let  $U_\alpha$  be a rotation by angle  $\alpha$ 
4 // check classically
5 if  $x \notin \{a, b\}^* c \{a, b\}^*$  then
6   | Reject ;
7 end
8 while true do
9   | // from  $\dagger$  to ' $c$ '
10  for each symbol  $\sigma$  do
11    |  $q = U_{\sqrt{2}\pi} q$  ;
12  end
13  | // from ' $c$ ' to '$'
14  for each symbol  $\sigma$  do
15    |  $q = U_{-\sqrt{2}\pi} q$ ;
16  end
17  Measure quantum state ;
18  if  $q = |q_1\rangle$  then
19    | Reject ;
20  end
21  else if simulate probability  $\frac{1}{2^k l^2}$  then
22    | Accept ;
23  end
24 end

```

Algorithm 15: L_m 2QCFA

Language Classification: Context-Free. **Error Type:** One-sided.

Language L_p is another simple extension of $L_=_$; therefore, it is natural that a variation of $M_{L_=_}$ is able to recognize it. It suffices to calibrate the rotation of the symbol ‘ b ’ by p times its value in $M_{L_=_}$.

For a word $x \in L$, the probability of correctly recognizing it is still one. However, when $x \notin L$, the probability p_{rej} of rejecting at each iteration is now lower by a constant factor $\frac{1}{p^2}$, resulting in

$$p_{rej} \geq \frac{1}{2(pl)^2}.$$

The factor $\frac{1}{p^2}$ is constant, so it is enough to calibrate the error constant, k , to achieve the desired error ϵ .

The algorithmic description of the 2QCFA M_{L_p} is given in 25.

Theorem 11.3.12. *The 2QCFA M_{L_p} in 25 recognizes L_p with one-sided error ϵ .*

Proof. Analogous to the proof of 11.3.3 for $L_=_$. □

11.3.5 Language $L_{a^n b^n c^n}$

The context-sensitive language $L_{a^n b^n c^n}$ is defined below.

Definition 11.3.13.

$$L_{a^n b^n c^n} = \{a^n b^n c^n | n \in N\}.$$

Language Classification: Context-Sensitive. **Error Type:** One-sided.

The language $L_{a^n b^n c^n}$ is not a context-free language, but it is possible to recognize it using at most a linear amount of memory. Consequently, it is a context-sensitive language. This language can be expressed as the intersection of two context-free languages, that is, $L_{a^n b^n c^n} = \{a^* b^n c^n | n \in N\} \cap \{a^n b^n c^* | n \in N\}$. The intersection operation is not closed for CFL class, but it is closed for the 2QCFA model. It is easy to imagine an automaton based on $M_{L_=_}$, to recognize each of those CFLs. For the language $\{a^n b^n c^* | n \in N\}$, it suffices to ignore the symbol c , and proceed the verification of $a^n b^n$. The language $\{a^* b^n c^n | n \in N\}$ is symmetric. For this reason, we omit the algorithmic description of the 2QCFA $M_{L_{a^n b^n c^n}}$ for $L_{a^n b^n c^n}$. Instead, we simply state the theorem.

Theorem 11.3.14. *There is a 2QCFA $M_{L_{a^n b^n c^n}}$ that recognizes $L_{a^n b^n c^n}$ with one-sided error ϵ .*

```

1  Let  $x$  be the input word
2  Let  $l$  be the length of  $x$ 
3  Let  $U_\alpha$  be a rotation by angle  $\alpha$ 
4  // check classically
5  if  $x \notin a^*b^*$  then
6  |   Reject ;
7  end
8  while true do
9  |   // from † to $
10 |   for each symbol  $\sigma$  do
11 |       if  $\sigma = a$  then
12 |           |    $q = U_{\sqrt{2}\pi}q$  ;
13 |       end
14 |       else
15 |           |    $q = U_{-p\sqrt{2}\pi}q$  ;
16 |       end
17 |   end
18 |   Measure quantum state ;
19 |   if  $q = |q_1\rangle$  then
20 |       |   Reject ;
21 |   end
22 |   else if simulate probability  $\frac{1}{2^kl^2}$  then
23 |       |   Accept ;
24 |   end
25 end

```

Algorithm 16: L_p 2QCFA

11.3.6 Languages $L_>$ and $L_<$

Languages $L_>$ and $L_<$ are another natural extension of $L_=$, in which the number of a 's is greater than the number of b 's, and vice-versa, respectively.

Definition 11.3.15.

$$L_> = \{a^i b^j \mid i, j \in N \text{ and } i > j\}.$$

$$L_< = \{a^i b^j \mid i, j \in N \text{ and } i < j\}.$$

Error Type: Two-sided. **Language Classification:** Context-Free.

Theorem 11.3.16. *For any $\epsilon > 0$, there is a 2QCFA, $M_{L_>}$, that recognizes $L_>$ with error ϵ .*

To prove this theorem, we use ideas developed in [7] and [81]. We know that there is a 2QCFA, $M_=$, that recognizes $L_=$ with one-sided error $\epsilon_=$. $M_=$ will be simulated by $M_{L_>}$, so that we can discard with certainty $a^n b^n$. Afterwards, we try to detect if there are more 'a's or more 'b's by flipping coins. Let m be the number of 'a's and n the number of 'b's. We are looking for the side (*i.e* the one with 'a's or 'b's) whose outcome is all heads, for k rounds. The side that is shorter will win most of the time. If the 'b' side wins, there is a high probability that the word belongs to $L_>$, and $M_{L_>}$ accepts it. Otherwise, if the 'a' side wins, $M_{L_>}$ rejects it. Its also possible that no side flips all heads in k rounds, in this case the process is repeated until a side wins.

The algorithmic description of the 2QCFA $M_{L_>}$ that recognizes $L_>$ is given in 17.

Proof. Let x be the input word, m be the number of 'a's, and n be the number of 'b's. If x is not in the form $a^m b^n$, it is readily rejected at the first **if**. If $x \in L_>$, there is an $\epsilon_=$ chance that $M_=$ will accept it, making $M_{L_>}$ reject it. The $\epsilon_=$ can be made as close to zero as possible by running $M_=$ multiple times. If $M_=$ rejects x we are sure that $m \neq n$, because $M_=$ has an one-sided error. In this case, we analyse the trivial cases in which there is just one 'a' or just one 'b'. From this point on, we have $m, n \geq 1$. The **while** loop has probability p_a of accepting x at each iteration

$$p_a = \left[\left(1 - \frac{1}{2^m}\right) \frac{1}{2^n}\right]^k.$$

The probability of rejecting x at each iteration is p_r

$$p_r = \left[\left(1 - \frac{1}{2^n}\right) \frac{1}{2^m}\right]^k.$$

The probability that it will not halt is $(1 - (p_a + p_r))$. Therefore, the probability of eventually accepting x is

```

1  Let  $M_{=}$  be a 2QCFA of  $L_{=}$ 
2  if  $x \notin a^*b^*$  then
3    | Reject ;
4  end
5  if  $M_{=}$  accepts  $x$  then
6    | // here we may reject words from  $L_{>}$  with probability  $\epsilon_{=}$ ,
7    | // and we also reject all words from  $L_{=}$ 
8    | Reject ;
9  end
10 if  $x = a$  then
11   | Accept ;
12 end
13 else if  $x = b$  then
14   | Reject ;
15 end
16 // At this point,  $x$  is in the form  $a^mb^n$  with  $m, n \geq 1$  and  $m \neq n$ 
17 while True do
18   | // Check which value is greater  $m$  or  $n$ 
19   for  $i = 1..k$  do
20     |  $event_1^i$  = Check if  $m$  coin flips are all heads;
21     |  $event_2^i$  = Check if  $n$  coin flips are all heads;
22   end
23   if  $event_j^i$  is all heads for  $i \in \{1, \dots, k\}, j \in \{1, 2\}$  then
24     | // A draw, so try again
25     | Continue ;
26   end
27   else if  $event_1^i$  is all heads for  $i \in \{1, \dots, k\}$  then
28     | //  $m$  may be smaller than  $n$ 
29     | Reject ;
30   end
31   else if  $event_2^i$  is all heads for  $i \in \{1, \dots, k\}$  then
32     | //  $n$  may be smaller than  $m$ 
33     | Accept ;
34   end
35 end

```

Algorithm 17: $L_{>}$ 2QCFA

$$P_a = \sum_{i \geq 0} (1 - (p_a + p_r))^i p_a.$$

Using the sum of a convergent geometric progression, we have

$$P_a = \frac{1}{1 + \frac{p_r}{p_a}}.$$

Analogously, the overall probability of rejection is

$$P_r = \frac{1}{1 + \frac{p_a}{p_r}}.$$

Since $m \neq n$, the ratio, $\frac{p_a}{p_r}$ can be bounded. Suppose that $m > n$ (the other case is analogous).

$$\frac{p_a}{p_r} = \left[\frac{2^m - 1}{2^n - 1} \right]^k \geq \left[\frac{2^{p+1} - 1}{2^p - 1} \right]^k \geq \left[\frac{2 - \frac{1}{2^p}}{1 - \frac{1}{2^p}} \right]^k \geq \left[\frac{2}{1} \right]^k$$

If $x \in L_{>}$, we can make P_a as close to 1 as we want, and, at the same time, P_r will be close to 0. If $x \notin L_{>}$, due to the symmetry in the probabilities P_a and P_r , the opposite occurs. Let $\epsilon_{>}$ be the error probability in the loop. Once both probabilities $\epsilon_{>}$ and $\epsilon_{=}$ can be arbitrarily approximated to zero, and the $M_{L_{>}}$ error is at most $\epsilon \leq \max\{\epsilon_{=}, \epsilon_{=} + \epsilon_{>} - \epsilon_{=}\epsilon_{>}\}$, the theorem holds.

Case Analysis

We now give a thorough case analysis for a generic input x .

- x is not in the form $a^m b^n$: rejected with probability 1;
- $x \in L_{=}$: rejected with probability 1, since $M_{=}$ accepts $L_{=}$ with probability 1;
- $x \in L_{>}$:
 - $M_{=}$ may accept x with probability $\epsilon_{=}$, so $M_{L_{>}}$ rejects x with probability $\epsilon_{=}$;
 - $M_{=}$ rejects x with probability $1 - \epsilon_{=}$:
 - * $M_{L_{>}}$ accepts x with total probability $(1 - (\epsilon_{=} + \epsilon_{>} - \epsilon_{=}\epsilon_{>}))$;
 - * $M_{L_{>}}$ rejects x with total probability $\epsilon_{=} + \epsilon_{>} - \epsilon_{=}\epsilon_{>}$.
- $x \in L_{<}$: the result is analogous to the case $x \in L_{>}$.

The total error of $M_{L_{>}}$ is then $\epsilon = \max\{\epsilon_{=}, \epsilon_{=} + \epsilon_{>} - \epsilon_{=}\epsilon_{>}\} \leq \epsilon_{>} + \epsilon_{=}$.

□

11.3.7 The 2QCFA and the Chomsky Hierarchy

Currently, there is no example of a language that the 2QCFA can not recognize. Its original definition does not enforce a limit on the precision of the unitaries it may use. For this reason, it is not ruled out that it can recognize all languages (**ALL**). It is evident that considering the 2QCFA in practice would require imposing such a limit. Focusing only on the known acceptance languages, this model relative to the Chomsky Hierarchy can be depicted as in the Figure 11.1.

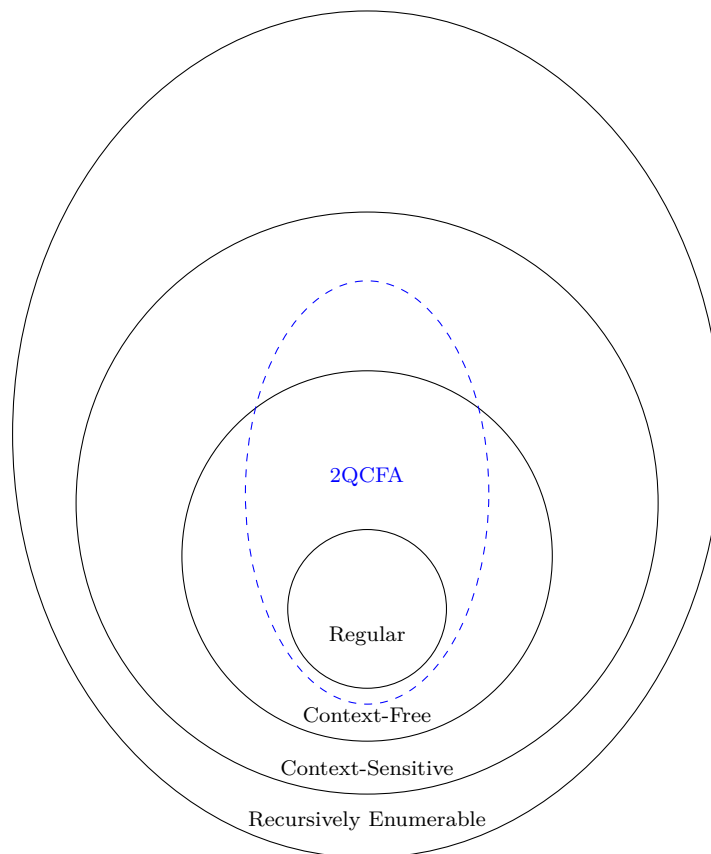


Figure 11.1: Tentative Position of 2QCFA in the Chomsky Hierarchy.

11.4 Closure Properties

In this section, we state known closure properties of the 2QCFA model, and give informal proofs for them. These properties were introduced in [82].

11.4.1 Union

Theorem 11.4.1 (Union). *Let L_1 and L_2 be two languages recognizable by a 2QCFA with error ϵ_1 and ϵ_2 , respectively. The language $L = L_1 \cup L_2$ is also recognizable by a 2QCFA with error $\epsilon = \epsilon_1 + \epsilon_2 - \epsilon_1\epsilon_2$.*

Proof. Let M_1 , M_2 , and M be 2QCFA accepting L_1 , L_2 , and L , respectively. Firstly, M simulates M_1 on the input word x . If M_1 accepts x , then M accepts it. Otherwise, M simulates M_2 . If M_2 accepts x , then M accepts it. If M_2 rejects x , M also rejects it. To determine M 's error, we have to analyse some cases. If $x \in L_1$, but $x \notin L_2$, M accepts x with probability at least $1 - \epsilon_1$. In the case of $x \in L_2$, but $x \notin L_1$, M accepts x with probability at least $(1 - \epsilon_1)(1 - \epsilon_2) + \epsilon_1$. That is the probability of M_1 correctly rejecting x , and M_2 correctly accepting x , or the probability of M_1 being wrong. When $x \in L_1 \cap L_2$, the probability of accepting x is at least $1 - \epsilon_1$. If $x \notin L$, the probability of rejecting it is $(1 - \epsilon_1)(1 - \epsilon_2)$, since both M_1 and M_2 reject it. The largest of these error probabilities is $\epsilon_1 + \epsilon_2 - \epsilon_1\epsilon_2$, and it happens when both, M_1 and M_2 , must reject x . Therefore, M 's error is $\epsilon = \epsilon_1 + \epsilon_2 - \epsilon_1\epsilon_2$. \square

11.4.2 Intersection

Theorem 11.4.2 (Intersection). *Let L_1 and L_2 be two languages recognizable by a 2QCFA with error ϵ_1 and ϵ_2 , respectively. The language $L = L_1 \cap L_2$ is also recognizable by a 2QCFA with error $\epsilon = \epsilon_1 + \epsilon_2 - \epsilon_1\epsilon_2$.*

Proof. This proof is very similar to the union closure proof. Let M_1 , M_2 , and M be the 2QCFA for L_1 , L_2 , and L , respectively. Firstly, M simulates M_1 on the input x , if M_1 rejects it, then M also rejects it. Otherwise, M simulate M_2 on x , if M_2 accepts it, then M also accepts it. If M_2 rejects x , M also rejects it. Once again, we have to analyse each case. If $x \in L$, the probability of correctly accepting x is $(1 - \epsilon_1)(1 - \epsilon_2)$, since both M_1 and M_2 must accept it. If $x \notin L_1$, the probability of rejecting it is, at least, $1 - \epsilon_1$. If $x \in L_1$, but $x \notin L_2$, the probability of rejecting it is, at least, $(1 - \epsilon_1)(1 - \epsilon_2) + \epsilon_1$, that is if M_1 wrongly rejects x , or if M_1 accepts it and M_2 rejects it. The largest of these error probabilities is $\epsilon_1 + \epsilon_2 - \epsilon_1\epsilon_2$, and it happens when both, M_1 and M_2 , must accept x . Therefore, M 's error is $\epsilon = \epsilon_1 + \epsilon_2 - \epsilon_1\epsilon_2$. \square

11.4.3 Complement

Theorem 11.4.3 (Complement). *Let L be a language recognizable by a 2QCFA with error ϵ . The language $\bar{L} = \Sigma^* \setminus L$ is also recognizable by a 2QCFA with error ϵ .*

Proof. Let M be a 2QCFA that recognizes L . The 2QCFA \overline{M} for \overline{L} can be build from M just by transforming the accepting states into rejecting states, and vice versa. For an input word $x \in L$, the probability of \overline{M} rejecting it is, at least, $1 - \epsilon$. If $x \notin L$, the probability of accepting it is, at least, $1 - \epsilon$. \square

11.4.4 Reversal

Theorem 11.4.4 (Reversal). *Let L be a language recognizable by a 2QCFA with error ϵ . The language $L^R = \{x^r | x \in L\}$ is also recognizable by a 2QCFA with error ϵ .*

Proof. Let M be a 2QCFA for L . A 2QCFA for L^R can be obtained by reversing the original movements of M . \square

11.4.5 Special Concatenation

Theorem 11.4.5 (Special Concatenation). *Let L_1 and L_2 be two languages recognizable by a 2QCFA with error ϵ_1 and ϵ_2 , respectively. In addition, suppose that the alphabets of L_1 and L_2 are disjoint. The language $L = \{xy | x \in L_1 \text{ and } y \in L_2\}$ is also recognizable by a 2QCFA with error $\epsilon = \epsilon_1 + \epsilon_2 - \epsilon_1\epsilon_2$.*

Proof. In the general case of concatenation, there is no indication where the subword $y \in L_2$ starts in the input word. However, when the alphabets of L_1 and L_2 are disjoint, this problem no longer exists. Let M_1 , M_2 , and M be 2QCFA for L_1 , L_2 , and L , respectively. When M simulates M_1 , it treats any symbols from L_2 's alphabet as $\$$: the end marker. Analogously, when it simulates M_2 it treats any symbol from L_1 's alphabet as \dagger : the starting marker. Now, we have two substrings to check, and the remainder of the proof is very similar to the intersection case, when both automaton must accept it. Therefore, M 's error is $\epsilon = \epsilon_1 + \epsilon_2 - \epsilon_1\epsilon_2$. \square

11.5 Discussion

Currently, we only know some examples of languages recognizable by the 2QCFA model. However, there is neither a characterization for this class of languages, such as a pumping lemma, nor we have a concrete proof of a language not recognizable by this model. These are two important open question for future research. Potentially, fixed dimensional quantum memory can store an infinite amount of information since amplitudes are complex numbers. Therefore, it is challenging to determine what a 2QCFA cannot compute.

Chapter 12

On the Quantum-Classical Separation of Marking and Multi-Head Automata

12.1 Introduction

Feynman introduced the Quantum Computing paradigm after realizing that simulating quantum physics on classical computers was seemingly hard [36]. Since then, many quantum computational models have been proposed. A crucial question is whether these models are more powerful in terms of computability and computational complexity than their classical analogues. Quantum-classical separation theorems are important tools to investigate these questions. In this chapter, we introduce a new quantum automata model which uses markers and is based on the Two-way Quantum Classical Finite Automaton (2QCFA) formalism of Ambainis and Watrous [7]. Moreover, for any given number of markers we establish a new result that this new model is more powerful than the classical marking extensions of the Two-way Deterministic Finite Automaton (2DFA) and the Two-way Probabilistic Finite Automaton (2PFA) in terms of computability and complexity, respectively.

Several models of quantum automata have been proposed [66, 72, 99, 103]. The 1QFA model recognizes only a proper subset of the class of all regular languages [66, 72]. Its bi-directional extension, the 2QFA model, is strictly more powerful in the sense that it recognizes even non-regular languages [46, 66]. The automaton model that is the base for this chapter, namely the 2QCFA, uses only constant classical and quantum memory and it is also strictly more powerful than the 2DFA model, regarding computability and complexity [7, 82]. It is known that it recognizes all regular, some non-regular, and even some non-context-free languages. Using the 2QCFA model it is possible to recognize

the palindrome language over a two letter alphabet, an impossible task even for a 2PFA with bounded error [32]. Besides, a 2QCFA can also recognize the non-regular language $L_+ = \{a^n b^n | n \in \mathbb{N}\}$ in polynomial time, whereas exponential time is needed in the 2PFA model.

There are several studies treating marking (pebble) and multi-head automata in the classical model [31, 52, 54, 78, 79, 102]. It is natural to study the impact of such extensions in the quantum paradigm, and investigate quantum-classical separations in these more general models. In this context, Zheng et al. proposed a multi-head automata based on the 2QCFA model [103]. They presented the language $L_{pow(k)}$ as an indication that their new model was more powerful than its classical analogue in terms of computability for every number k of heads, with $k \geq 3$. Moreover, they left open the question of whether this was actually true. In this chapter, we show that, in fact, that language can not be used to establish such a separation. On the other hand, using other techniques, we settle this question in the affirmative by proving that their multi-head quantum model is more powerful than its classical analogue for any given number of heads.

Yakaryilmaz introduced a one counter automata based on the 2QCFA model and showed a quantum-classical computability separation of this new model and the classical Two-way Deterministic Counter Automaton (2DCA), for sub-linear counter space [99, 100]. In principle, the counter space in this model can be arbitrarily large, but the languages studied in that work required only a logarithmic counter space. In such cases, replacing the counter by a pebble would be equivalent. In fact, this quantum counter automaton is a special case of our marking 2QCFA model with just one marker.

In Section 12.2, we introduce the k -M2QCFA model, a marking extension of the 2QCFA model, and we establish some general notation. In Section 12.3, we show that the language $L_{pow(k)}$ [103] can not be used to obtain a computability quantum-classical separation for any number k of heads, for $k \geq 3$. In Section 12.4, we present our main results regarding quantum-classical computability and complexity separations for marking and multi-head finite automata. In Section 12.5, we revisit and prove some 2QCFA error amplification lemmas that are used in Section 12.4, but that can also be of independent interest. Finally, in Section 12.6, we pose some open problems. We assume the reader is familiar classical automata theory [53].

12.2 Definitions and Notations

In this section, we define the k -M2QCFA model and establish some basic notation.

12.2.1 Multi Marker 2QCFA (k -M2QCFA)

A k -M2QCFA is constructed by taking the 2QCFA model of Ambainis and Watrous [7] and extending it with the ability to place at most k labeled markers in the input string. We define this model in the same way as was done for the classical automata model [54]. Formally, a k -M2QCFA is defined by a 10-tuple, $M = (Q, S, \Sigma, \Theta, \delta, K, q_0, s_0, S_{acc}, S_{rej})$, where:

- Q and S are sets of quantum and classical states, respectively;
- Σ is the input alphabet;
- Θ and δ are the quantum and classical evolution functions, respectively;
- $K = \{1, \dots, k\}$ is the set of labeled markers;
- q_0 and s_0 are the quantum and classical initial states, respectively;
- S_{acc} and S_{rej} are sets of classical accepting and rejecting states, respectively, with $S_{acc} \cap S_{rej} = \emptyset$.

The input tape has two tracks, namely, the input track and the marker track. Both tracks have the same number of cells. The tape head reads simultaneously the current symbol from the input and the marker tracks.

Now, we describe the input track. A generic input word x is placed in the input track with two special delimiters, \dagger and $\$$, to mark the beginning and the end of input, respectively. These two symbols are not part of the input alphabet; and the track alphabet, Γ , is defined as $\Gamma = \{\dagger, \$\} \cup \Sigma$. Track position 0 contains the symbol \dagger . For $i \in [1, |x|]$, track position, i , contains symbol x_i where $x = x_1x_2 \dots x_{|x|}$. Finally, track position $|x| + 1$ contains the symbol $\$$. The automaton is not allowed to move the tape head to the left of \dagger , nor to the right of $\$$.

For each cell in the input track, there is a corresponding cell in the marker track. The marker track has the set $K' = K \cup \{0\}$ as its alphabet. It is initialized with all 0s, which indicates the absence of markers. For any $m \in K$, there must be no more than one occurrence of m in the marker track. In spite of each cell holding only one marker, this is just a notational simplification, since it is possible to store in the automaton states which markers share the same position of another already placed marker, as the number of markers is a fixed constant. Let $S_{halt} = S_{acc} \cup S_{rej}$ be the set of halting states, and $S_{non} = S \setminus S_{halt}$ be the set of non halting states. We denote by s and q be the current classical and quantum states, respectively. Moreover, let σ and m be the current symbol in the input track and the current marker in the marker track.

The computation of a k -marker 2QCFA starts with the machine in states q_0 and s_0 . At each iteration, the automaton uses s and σ to determine the quantum evolution $\Theta(s, \sigma)$, which can be a unitary transformation or a measurement. Formally, Θ is a mapping

$$\Theta : S_{non} \times \Gamma \rightarrow \mathcal{U}(\mathcal{H}(Q)) \cup \mathcal{M}(\mathcal{H}(Q)).$$

The set $\mathcal{U}(\mathcal{H}(Q))$ contains unitary transformations in the fixed dimension Hilbert space whose base states are in Q . Analogously, the set $\mathcal{M}(\mathcal{H}(Q))$ contains projective measurements in the same space. Next, the classical evolution δ is applied. Its form depends on the type of quantum evolution previously performed. If it was a measurement, then δ is the mapping $\delta : S_{non} \times \Gamma \times K' \times R \rightarrow S \times K' \times D$, where R is the set of possible measurement results, K is the set of markers, and $D = \{-1, 0, +1\}$ is the set of tape head directions of movement. The tape head directions left, none, and right are mapped to -1 , 0 , and $+1$, respectively. Otherwise, if the last quantum operation was a unitary transformation, δ is the usual classical transition function of the form $\delta : S_{non} \times \Gamma \times K' \rightarrow S \times K' \times D$.

The computation continues until a halting state is reached in which case the automaton accepts if $q \in S_{acc}$, and it rejects otherwise. For a given word x , there is a probability p_{acc} of x being accepted and a probability p_{rej} of it being rejected. Since the automaton may not halt, $p_{acc} + p_{rej}$ may be smaller than 1.

We observe that labeled and non-labeled markers are equivalent in terms of computability. This is because the number of markers is constant, and so a non-labeled M2QCFA can store in its classical states the current permutation of markers placed on the marker tape, together with the two markers on the left and on the right of the reading head. Since non-labeled markers are equivalent to pebbles, a k -M2QCFA is also a pebble automaton.

12.2.2 Notation

We present some definitions and notations used throughout this chapter, and we review some known quantum and classical automaton models.

Definition 12.2.1 ($val[\cdot]$, $pos[\cdot]$). *Let $x = \sigma_1\sigma_2 \dots \sigma_l$ ($l \geq 0$) be a word over the alphabet Σ . Let i be an integer value between 1 and l . We denote by $val[x, i]$ the symbol at the i^{th} position, that is, $val[x, i] = \sigma_i$. Further, let h be the input head and suppose it is scanning a symbol σ_i . We define $val[x, h] = \sigma_i$. Let m be the marker under the corresponding cell of σ_i . We denote by $pos[x, m]$ the position of m , that is, $pos[x, m] = i$. When x is clear from the context, we use the abbreviated notations $val[i] = val[x, i]$, $val[h] = val[x, h]$, and $pos[x, m] = pos[m]$.*

Definition 12.2.2 ($\Sigma[\cdot]$). *Let L be a language. We denote by $\Sigma[L]$ the alphabet of L .*

Definition 12.2.3 (*markers[.]*). Let M be a 2QCFA. We denote by $\text{markers}[M]$ the number of markers used by M . Note that we consider any M2QCFA also to be a 2QCFA, in the sense that the former is at least as powerful as the latter. If L is a language, $\text{markers}[M, L]$ is the minimum number of markers required to recognize L using model M . When M is clear from the context, we use $\text{markers}[L]$ to denote $\text{markers}[M, L]$.

Definition 12.2.4 (*0-sided[.], 1-sided[.], 2-sided[.], and $L[.]$*). Let M be a computational model, we denote by $0\text{-sided}[M]$, $1\text{-sided}[M]$, and $2\text{-sided}[M]$ the set of languages that can be recognized by M with zero error, one-sided error (words in the language always accepted [7]) and two sided error, respectively. Moreover, $L[M] = 0\text{-sided}[M] \cup 1\text{-sided}[M] \cup 2\text{-sided}[M]$.

Table 12.1 summarizes the automata models used in this chapter. We use a concatenation of an extension, a direction, a type, and the suffix **FA** to designate a finite automata model. An extension can be empty, **H** for Multi-head, **SH** for sensing Multi-head, and **M** for marking. A direction can be 1 or 2 for one-way and two-way, respectively, and concerns the allowed movements of the automaton head. A type can be **D** for deterministic, **P** for probabilistic, and **QC** for quantum classical.

Model	Description
2DFA	Two-way Deterministic FA [53]
2PFA	Two-way Probabilistic FA [26]
2QCFA	Two-way Quantum Classical FA [7]
M2DFA	Marking Two-way Deterministic FA [54]
H2DFA	Multi-head Two-way Deterministic FA [52]
SH2DFA	Sensing Multi-head Two-way Deterministic FA [52]
M2PFA	Marking Two-way Probabilistic FA
M2QCFA	Marking Two-way Quantum Classical FA
H2QCFA	Multi-head Two-way Quantum Classical FA [103]
SH2QCFA	Sensing Multi-head Two-way Quantum Classical FA

Table 12.1: Automata Summary.

12.3 $L_{pow(k)}$ cannot separate k -H2DFA from k -H2QCFA for every k

Zheng et al. [103] used the language $L_{pow(k)}$, see Definition 12.3.1, as an indication of the separation between non-sensing k -head 2DFA (k -H2DFA) and non-sensing k -head 2QCFA

(k -H2QCFA) for every $k \geq 3$. In this section, we show that this language cannot be used for this separation, since a 4-marker 2DFA (4-M2DFA) can recognize it, and a k -marker 2DFA can be simulated by a non-sensing $(k + 1)$ -H2DFA. Therefore, there is a non-sensing 5-H2DFA for accepting $L_{pow(k)}$ for every k . Firstly, we investigate the inclusion $L[k\text{-M2DFA}] \subseteq L[(k + 1)\text{-H2DFA}]$. We close this section presenting a description of a 4-M2DFA that accepts $L_{pow(k)}$.

Definition 12.3.1. *The power language, $L_{pow(k)}$, is defined as:*

$$L_{pow(k)} = \{a^n b^{n^k} \mid n > 1 \text{ and } k \text{ a fixed constant in } \mathbb{N}\}.$$

12.3.1 $(k + 1)$ -head 2DFA can simulate k -marker 2DFA

It is already known that a $(k + 1)$ -H2DFA can simulate k -M2DFA [78]. We just restate this result in terms of the equivalence between sensing k -SH2DFA and k -M2DFA [79].

Theorem 12.3.2. $L[k\text{-M2DFA}] \subseteq L[\text{non-sensing } (k + 1)\text{-H2DFA}]$.

Proof. Let A be a k -M2DFA. From [79], there is a sensing k -SH2DFA B that simulates A . Note that this result is not trivial, since as at least one head of B must simulate A 's head leaving in at most $k - 1$ heads to simulate the markers. Now, it suffices to show how B can be simulated by a non-sensing $(k + 1)$ -H2DFA C . In order to provide the sensing feature, we use an extra head, denoted h_e . Each time we want to know if two heads, h_i and h_j , are in the same cell we run the simple Algorithm 18. \square

12.3.2 4-M2DFA to accept $L_{pow(k)}$, for any k

Zheng et al. created a non-sensing $(k + 1)$ -H2DFA for $L_{pow(k)}$. They used k heads to store digits of a number in base n . We follow a different approach. We show that given two delimited regions — two contiguous sequence of symbols — in the input word, of sizes C_1 and C_2 , it is possible to obtain a delimited region comprising $C_1 C_2$ symbols, or it is possible to detect no such region exists, using only a fixed number of markers. This is formalized in the next lemma.

Lemma 12.3.3. *Let $x = a^+ b^+$ be an input word, and let $l = |x|$. Let n denote the number of a 's in x . If we have a delimited region from positions p_{start} to p_{end} of size n^k , it is possible to find a region of size n^{k+1} starting from p'_{start} , provided that $(p'_{start} + n^{k+1}) \leq l$. We assume that it is possible to detect p_{start} , p_{end} , and p'_{start} . This procedure requires three additional temporary counters, and it runs in polynomial time. Moreover, the case $(p'_{start} + n^{k+1}) > l$ can be detected.*


```

1 Place head  $h_e$  at  $\dagger$  ;
2  $same = yes$ ; //  $h_i$  and  $h_j$  at same position?
3 while  $val[h_i] \neq \dagger$  and  $val[h_j] \neq \dagger$  do
4   | Move  $h_e$  one position to the right ;
5   | Move  $h_i$  and  $h_j$  one position to the left ;
6 end
7 if  $val[h_i] \neq val[h_j]$  then
8   |  $same = false$  ;
9 end
10 // Restore  $h_i$  and  $h_j$  original positions
11 while  $val[h_e] \neq \dagger$  do
12   | Move  $h_e$  one position to the left ;
13   | Move  $h_i$  and  $h_j$  one position to the right ;
14 end

```

Algorithm 18: Sensing feature with one extra head.

Proof. The idea is to use the region of a 's that contains n symbols in order to multiply the number of symbols between p_{start} and p_{end} by n .

Let m_{count} be a marker that will range from p_{start} to p_{end} . Let $m_{prospect}$ be a prospective marker that will start at p'_{start} and will finish at $(p'_{start} + n^{k+1})$, provided that x is large enough. Let m_a be a marker that will be used in the region of a 's. For each symbol within p_{start} to p_{end} , we do the following: starting from m_a in the first a ', we advance m_a and $m_{prospect}$ one position to the left until $val[pos[m_a]] = b$. Algorithm 19 makes this idea precise. Each step of the inner while loop adds n symbols to the region delimited by p'_{start} and $m_{prospect}$. Since this process is repeated n^k times, we end up with n^{k+1} symbols.

Placing markers m_{count} , $m_{prospect}$, and m_a requires time $O(l)$. Each step of the inner while loop takes time $O(l)$ and is executed at most $O(l)$ times. The outer while loop is executed $O(l)$ times. Therefore, the total running time is $O(l^3)$.

If the input x is shorter than $(p'_{start} + n^{k+1})$, this situation is detected when $m_{prospect}$ is advanced. \square

The following lemma shows how to find a region of size n^k . It is basically an induction using Lemma 12.3.3.

Lemma 12.3.4. *Let $x = a^+b^+$ be an input word, and let $l = |x|$. Let n and num_b denote the number of a 's and b 's in x , respectively. If $num_b \geq n^k$, then it is possible to find a region of size n^k starting at the first b '. Otherwise, it is possible to detect that $num_b < n^k$. This procedure runs in polynomial time and uses at most 4 markers.*

Proof. We proceed by induction on $k \geq 1$. Let $k = 1$. In this case, we use two markers m_a and $m_{prospect}$. Marker m_a starts from the first a and $m_{prospect}$ starts from the first b .

```

1  Let  $p_{start}$  and  $p_{end}$  be the positions delimiting a region of size  $n^k$  ;
2  Place a marker  $m_{count}$  at  $p_{start}$  ;
3  Place a marker  $m_{prospect}$  at  $p'_{start}$  ;
4  while  $pos[m_{count}] \leq p_{end}$  do
5      Place a marker  $m_a$  at the first 'a';
6      while  $val[pos[m_a]] \neq b$  do
7          Move  $m_a$  one position to the right ;
8          Move  $m_{prospect}$  one position to the right ;
9          //  $|x| < p'_{start} + n^{k+1}$ 
10         if  $val[pos[m_{prospect}]] = \$$  then
11             Error ;
12         end
13     end
14     Move  $m_{count}$  one position to the right ;
15 end

```

Algorithm 19: Multiply algorithm

Markers m_a and $m_{prospect}$ are advanced one position until $val[m_a + 1] = b$. The region from the first b to $m_{prospect}$ contains n symbols. In this step, we use only two markers, and it runs in time $O(l^2)$.

Inductive step. We assume the result holds for some k , with $k > 1$. Let p_{start} be the position of the first b from left to right. Let p_{end} be the position of a marker satisfying $(p_{end} - p_{start} + 1) = n^k$. From Lemma 12.3.3, making $p'_{start} = p_{start}$, the region can be increased to n^{k+1} using three additional markers. Since we were already using a marker for p_{end} , and p_{start} is implicit, at most 4 markers were used. For the running time, each of the k steps runs in $O(l^3)$. Therefore, the total running time is polynomial, even if $k \in O(p(l))$ for some polynomial $p(l)$. \square

We compile these results in the following theorem.

Theorem 12.3.5. *Let $x = a^+b^+$ be an input string, and let $l = |x|$. There is a 4-marker 2-way deterministic finite automaton that decides $L_{power(k)}$ in polynomial time.*

Proof. If the number of b s satisfies $num_b \geq n^k$, then Lemma 12.3.4 states that a region of n^k b s can be delimited. It suffices to ensure that $val[m_{prospect} + 1] = \$$ to guarantee that $x \in L_{power(k)}$. If $num_b < n^k$, by Lemma 12.3.4 this situation can be detected. For the running time, this same Lemma guarantees that the procedure runs in polynomial time, and detecting if $pos[m_{prospect}] = l$ requires $O(l)$ time. Therefore, the result follows. \square

12.4 Computability Separation

In this section, we prove a separation in the computability power of k -M2DFA and of k -M2QCFA, in the sense that there are languages recognizable by the latter but not by the former. We also prove a similar result for the multi-head case, by showing a computability separation between k -H2DFA and k -H2QCFA. This last separation result was left as an open question in [103]. Furthermore, we present a complexity separation between k -M2QCFA and k -M2PFA. To accomplish these results, we use a hierarchy of languages which was already used to prove that $(k+1)$ markers are better than k markers in the classical case [54]. This hierarchy is defined next:

Definition 12.4.1. $L_k(L)$ is a language defined inductively in terms of L as follows:

- If $k = 0$, then $L_0(L) = L$;
- Otherwise,

$$L_k(L) = \{x_1\#x_2\#\dots\#x_{2n} \mid \text{exactly half } x_i \in L_{k-1}(L), n \geq 1, \text{ and } \# \notin \Sigma[L_{k-1}(L)]\}.$$

Since half of the sub-strings x_i belong to $L_{k-1}(L)$, an automaton for deciding $L_k(L)$ will need to count. But, as n can be arbitrary, this counting is impossible when using only finite classical memory, and so a new marker will be required to implement the counting.

One important property of this language hierarchy is stated in the next theorem, adapted from [54].

Theorem 12.4.2. *If L requires m markers to be recognized, then $L_k(L)$ requires $m + k$ markers to be decided, using the M2DFA model.*

The next definition names the set of languages recognized by a k -M2DFA.

Definition 12.4.3. *The Classical Hierarchy level k , denoted by $CH(k)$, is defined as the set of languages $CH(k) = \{L \mid \text{there is a } k\text{-M2DFA that recognizes } L\}$.*

As a corollary to Theorem 12.4.2, we have a computability separation for 2DFA with $k+1$ and with k markers.

Corollary 12.4.4. $CH(k+1) \not\subseteq CH(k)$.

Proof. From Theorem 12.4.2, $L_{k+1}(L) \in CH(k+m+1)$. But $L_{k+1}(L) \notin CH(k+m)$ where $m = \text{markers}[L_k(L)]$. \square

As in the classical case, we define the set of languages recognized by a k -M2QCFA.

Definition 12.4.5. *The Quantum Hierarchy level k , denoted by $QH(k)$, is defined as the set of languages $QH(k) = \{L \mid \text{there is a } k\text{-M2QCFA that recognizes } L\}$.*

12.4.1 Quantum-Classical Separations

In order to prove quantum-classical separations, we first prove several auxiliary lemmas. Let L be a language, and consider languages $L_k(L)$ and $L_{k-1}(L)$ for some $k \geq 1$. Let M be a 2QCFA that accepts $L_{k-1}(L)$ with a two-sided error ϵ . For an input word $y = x_1 \# x_2 \# \dots \# x_{2n}$, the value of n can be arbitrarily large. Since the error ϵ is constant, the probability of correctly recognizing all x_i decreases exponentially if $\epsilon > 0$. The challenge is to amplify the error ϵ to $\frac{\epsilon}{f(n)}$ when simulating M on each x_i . We observe that the amplification must consider the global parameter of the input n ; it is not possible to analyze each x_i isolatedly, if $\epsilon > 0$.

The first lemma assumes that this amplification is possible. We denote by $\text{amplify}[M]$ the new 2QCFA recognizing the same language as M , but with error $\frac{\epsilon}{f(n)}$. With this assumption, we show how to create a 2QCFA M' that decides $L_k(L)$. The next two lemmas show how to amplify the error in the special cases of a one-sided and two-sided error 2QCFA, respectively. They use results from general amplification lemmas to be established in Section 12.5. These three lemmas will allow us to show that languages at levels 0, 1, and 2 can be recognized using the same number of markers in the quantum M2QCFA model. As a M2DFA is trivially a k -M2QCFA, we will be able to conclude that M2QCFA is computationally more powerful than its classical counterpart.

Lemma 12.4.6. *Let $L_k(L)$ and $L_{k-1}(L)$ be languages at two consecutive levels in the language hierarchy, based on a language L , and let $k \geq 1$. If there is a 2QCFA M for $L_{k-1}(L)$ such that $\text{amplify}[M]$ has a two-sided error $\frac{\epsilon_0}{f(n)}$, with $f(n) = cn^3$, for some constant c , then there is a 2QCFA M' that accepts $L_k(L)$.*

Proof. Automaton M' is constructed as described in Algorithm 20.

Let the input word be $y = x_1 \# x_2 \# \dots \# x_{2n}$. The probability of correctly recognizing all $2n$ x_i s is $(1 - \frac{\epsilon_0}{f(n)})^{2n}$. Using Bernoulli's inequality, this probability, denoted p_{suc} , can be bounded by:

$$p_{suc} \geq 1 - 2n \frac{\epsilon_0}{f(n)} . \quad (12.1)$$

Let $p'_{suc} = 1 - 2n \frac{\epsilon_0}{f(n)}$ be the minimum value of p_{suc} . Furthermore, let $n_{L_{k-1}(L)}$ and $n_{\neg L_{k-1}(L)}$ be the number of x_i terms in $L_{k-1}(L)$ but not in $L_{k-1}(L)$, respectively. In [7], Ambainis and Watrous determined the probability of detecting when the number of clockwise and counter-clockwise $\sqrt{2}\pi$ rotations differ. In other words, when $n_{L_{k-1}(L)} \neq n_{\neg L_{k-1}(L)}$ we get:

$$p_{detect} \geq \frac{1}{2(n_{L_{k-1}(L)} - n_{\neg L_{k-1}(L)})^2} . \quad (12.2)$$

```

1 // Let  $R_\alpha$  denote a rotation by an angle  $\alpha$ 
2 Check classically if the input is  $y = x_1\#x_2\#\dots\#x_{2n}$  ;
3 while True do
4   // Initialize one-qubit used for counting
5   Let  $q_{count} = q_0$  ;
6   for  $i \in \{1, \dots, 2n\}$  do
7     Simulate amplify[ $M$ ] on  $y$  with sub-string  $x_i$  ;
8     if  $M$  accepts then
9       |  $q_{count} = R_{\sqrt{2}\pi} q_{count}$  ;
10    end
11    else
12      |  $q_{count} = R_{-\sqrt{2}\pi} q_{count}$  ;
13    end
14  end
15  Measure  $q_{count}$  ;
16  if  $q_{count} = q_0$  then
17    // see the text for  $p'_{suc}$ 
18    if  $Rand[\frac{p'_{suc}}{2^k n^2}]$  then
19      | Accept ;
20    end
21  end
22  else
23    | Reject ;
24  end
25 end

```

Algorithm 20: Automaton M' accepting the language $L_k(L)$.

Since $n_{L_{k-1}(L)} + n_{\neg L_{k-1}(L)} = 2n$, p_{detect} can be bounded below by:

$$p_{detect} \geq \frac{1}{2(2n)^2} = \frac{1}{8n^2} . \quad (12.3)$$

If $y \notin L_k(L)$, the probability of rejecting at each iteration is:

$$p_{rej} \geq p'_{suc} \frac{1}{8n^2} . \quad (12.4)$$

If $y \in L_k(L)$, the probability of rejecting it at each iteration is $p_{rej} \leq (1 - p_{suc})p_{detect}$. Moreover, since $p_{detect} \leq 1$, we get:

$$p_{rej} \leq 2n \frac{\epsilon_0}{f(n)} 1 . \quad (12.5)$$

We fix the acceptance probability at each iteration to $p_{acc} = \frac{p'_{suc}}{2^k n^2}$ and show that the error of M' , denoted by ϵ' , can be made arbitrarily small.

If $y \in L_k(L)$, the total acceptance probability, denoted by P_{acc} , is:

$$P_{acc} = \sum_{i \geq 0} (1 - p_{acc} - p_{rej})^i p_{acc} = \frac{p_{acc}}{p_{acc} + p_{rej}} = \frac{1}{1 + \frac{p_{rej}}{p_{acc}}} . \quad (12.6)$$

Now, we show that the ratio $\frac{p_{rej}}{p_{acc}}$ can be made arbitrarily small when $y \in L_k(L)$. Let $\gamma = \frac{2n\epsilon_0}{f(n)}$. By adjusting $f(n)$, $1 - \gamma$ can be made greater than $\frac{1}{2}$. The ratio $\frac{p_{rej}}{p_{acc}}$ can be expressed as:

$$\frac{p_{rej}}{p_{acc}} \leq \frac{\gamma}{1 - \gamma} 2^k n^2 \leq \gamma 2^{k+1} n^2 . \quad (12.7)$$

For $f(n) = cn^3$, with c constant and $n > 1$, the fraction $\frac{p_{rej}}{p_{acc}}$ can be made arbitrarily small.

When $y \notin L_k(L)$, the total probability of rejecting, denoted P_{rej} , is:

$$P_{rej} = \sum_{i \geq 0} (1 - p_{acc} - p_{rej})^i p_{rej} = \frac{p_{rej}}{p_{rej} + p_{acc}} = \frac{1}{1 + \frac{p_{acc}}{p_{rej}}} . \quad (12.8)$$

In this case, the fraction $\frac{p_{acc}}{p_{rej}}$ can also be made arbitrarily small by adjusting k :

$$\frac{p_{acc}}{p_{rej}} \leq \frac{8n^2}{2^k n^2} . \quad (12.9)$$

The expected running time of M' is polynomial, provided that M runs in expected polynomial time. \square

The following lemma shows how a one-sided error 2QCFA for $L_{k-1}(L)$ can be amplified.

Lemma 12.4.7. *Let M be a 2QCFA accepting $L_{k-1}(L)$ with one-sided error ϵ . We build a 2QCFA M' with one-sided error at most $\frac{\epsilon}{f(n)}$, and that accepts $L_{k-1}(L)$. The expected number of iterations of M' is polynomial in n , provided that $f(n)$ is polynomial and M runs in expected polynomial time. The number of marker of M' is $\text{markers}[M'] = \max\{\text{markers}[M], 1\}$.*

Proof. (We will also denote M' by $\text{amplify}[M]$). Note that from the definition of $L_k(L)$ the parameter n is half the number of strings x_i in the input. The description of M' is given by Algorithm 21.

From Lemma 12.5.1 and given that $p_{stop} \leq \frac{1}{2f(n)\log f(n)}$, we have that $\text{amplify}[M]$ has a one-sided error of $\frac{\epsilon}{f(n)}$ and simulates M in a polynomial number of times in n . Moreover, the only marker $\text{amplify}[M]$ uses is never used when M is simulated. Therefore $\text{markers}[\text{amplify}[M]] = \max\{\text{markers}[M], 1\}$. \square

```

1 Let  $y = x_1 \# x_2 \# \dots \# x_{2n}$  ;
2 Let  $M$  be a 2QCFA with one-sided error  $\epsilon$ ;
3 Let  $p_{stop} = \frac{1}{2f(n)n}$  be the probability of stopping the procedure ;
4 Let the head be at  $pos_{x_i}$ : the position preceding the first symbol of  $x_i$  ( $\dagger$  or  $\#$ );
5 // To simulate probability  $p_{stop}$  we need to remember position  $x_i$ 
6 Place marker  $m_{x_i}$  at  $pos_{x_i}$  ;
7 //  $Rand[(1 - p_{stop})]$  returns true with probability  $(1 - p_{stop})$ 
8 while  $Rand[(1 - p_{stop})]$  do
9   //  $M$  is never simulated with marker  $m_{x_i}$  present
10  Remove marker  $m_{x_i}$  ;
11  Simulate  $M$  on  $x_i$  ;
12  if  $M$  rejects  $x$  then
13    | Reject ;
14  end
15  Place marker  $m_{x_i}$  at  $pos_{x_i}$  ;
16 end
17 Remove marker  $m_{x_i}$  ;
18 Accept ;

```

Algorithm 21: $amplify[M]$ where M is a one-sided error 2QCFA

The following lemma shows how a two-sided error 2QCFA for $L_{k-1}(L)$ can be amplified.

Lemma 12.4.8. *Let M be a 2QCFA accepting $L_{k-1}(L)$ with a two-sided error ϵ . We build a M2QCFA M' with two-sided error at most $\frac{\epsilon}{f(n)}$, and that accepts $L_{k-1}(L)$. The expected number of iterations of M' is polynomial in n , provided that $f(n)$ is polynomial and M runs in expected polynomial time. The number of markers of M' is $markers[M'] = \max\{markers[M] + 1, 2\}$.*

Proof. (We will also denote M' by $amplify[M]$). From Lemma 12.5.4, we know that if we can store and update a gambler's ruin game state starting with n coins for each player, then we can reduce the error ϵ to $\frac{\epsilon}{2f(n)}$, where $f(n)$ is $\cup_{k>1} O(k^n)$. Now, we show how to do this using a marker to store the game state. Initially, we use a one-sided 1-M2QCFA to find the n^{th} symbol $\#$ with probability $(1 - \frac{\epsilon}{2f(n)})$ and we place a marker m_{gamble} at this position. Note that we could have done this classically as we have two markers. However, we intend to keep the number of markers as low as possible. Afterwards, we simulate M on x_i . Each time it accepts x_i , we move m_{gamble} to the next $\#$, or $\$,$ to the right. Each time it rejects x_i we move m_{gamble} to the next $\#$, or $\dagger,$ to the left. If $val[m_{gamble}]$ is $\$,$ there is a high probability that $x_i \in L_{k-1}(L)$. Otherwise, there is a high probability that $x_i \notin L_{k-1}(L)$.

The description of M' is detailed in Algorithm 22. Note that $amplify[M]$ maintains

```

1  Let  $y = x_1 \# x_2 \# \dots \# x_{2n}$  ;
2  Let  $M$  be a 2QCFA with two-sided error  $\epsilon$ ;
3  Let the head be at  $pos_{x_i}$ : the position preceding the first symbol of  $x_i$  ( $\dagger$  or  $\#$ );
4  Place a marker  $m_{x_i}$  at  $pos_{x_i}$  ;
5  // Marker  $m_{gamble}$  will store the state of the gambler's ruin game
6  Place a marker  $m_{gamble}$  in the  $n^{\text{th}}$  symbol  $\#$  ;
7  while  $val[m_{gamble}] = \#$  do
8      | Remove  $m_{x_i}$  ;
9      | Run  $M$  on  $x_i$  ;
10     | Put  $m_{x_i}$  back at  $pos_{x_i}$  ;
11     | if  $M$  accepts  $x_i$  then
12         | Move  $m_{gamble}$  to the next  $\#$ , or to  $\$$  ;
13     | end
14     | else
15         | Move  $m_{gamble}$  to the previous  $\#$ , or to  $\dagger$  ;
16     | end
17 end
18 if  $val[m_{gamble}] = \$$  then
19     | Accept ;
20 end
21 else
22     | Reject ;
23 end

```

Algorithm 22: $amplify[M]$ where M is a two-sided error 2QCFA

the marker m_{gamble} while simulating M . Therefore, $amplify[M]$ use at least one more marker than M . Additionally, $amplify[M]$ uses an anchor marker m_{x_i} so that it does not loose the position of x_i . This marker m_{x_i} is not kept when simulating M , but it is used at the same time m_{gamble} is used. For this reason, the total number of markers used by M' is $markers[M'] = \max\{markers[M] + 1, 2\}$. \square

Now, we are ready to show the computability separations lemmas. Initially, we show how to go from a language L recognized by a M2DFA to a language $L_1(L)$ without using extra markers, by constructing a one-sided error 2QCFA. Secondly, we show how to go from a one-sided error 2QCFA for $L_1(L)$ to a two-sided 2QCFA for $L_2(L)$, again using the same number of markers. Finally, we show how to go from a two-sided error 2QCFA for $L_{k-1}(L)$ to a two-sided error M2QCFA for $L_k(L)$, but now we require one extra marker. These results imply that the language hierarchy collapses downward two levels in terms of the required markers.

Lemma 12.4.9. *If $L \in CH(\text{markers}[L])$, then $L_1(L) \in QH(\text{markers}[L])$. Moreover, $L_1(L)$ is accepted by a one-sided 2QCFA.*

Proof. Let M_0 be the m -M2DFA that recognizes L with $m = \text{markers}[L]$. We build M , a m -M2QCFA that simulates M_0 and has one-sided error. From our assumption, M_0 recognizes L with zero error as it is a M2DFA. Consequently, $\text{amplify}[M_0]$ satisfies the hypothesis of Lemma 12.4.6. For $y = x_1\#x_2\#\dots\#x_{2n} \in L_1(L)$, as every x_i is recognized exactly, the net rotation will be zero and M' will be a one-sided error automaton. \square

Lemma 12.4.10. *If $L \in CH(\text{markers}[L])$, then $L_2(L) \in QH(\text{markers}[L])$, for $\text{markers}[L] \geq 1$. Moreover, $L_2(L)$ is accepted by a two-sided 2QCFA.*

Proof. Lemma 12.4.9 states that there is a one-sided error 2QCFA M accepting L_1 . Besides, Lemma 12.4.6 states that from $\text{amplify}[M]$, with error $\frac{\epsilon}{f(n)}$, we can get a two-sided error 2QCFA accepting $L_k(L)$. From Lemma 12.4.7, we get this one-sided error amplification with the number of markers as $\min\{\text{markers}[M], 1\}$. \square

Lemma 12.4.11. *The language hierarchy shifts down two levels in the M2QCFA model, that is, $L_k(L) \in QH(\text{markers}[L] + k - 2)$ when $\text{markers}[L] \geq 2$.*

Proof. The proof is an induction on $k \geq 1$.

Cases $k = 1$ and $k = 2$ follow from Lemmas 12.4.9 and 12.4.10, respectively.

Suppose the result holds for some $k \geq 2$. Then there is a M2QCFA, M , with $\text{markers}[M] = (m + k - 2)$, $m \geq 2$, and $k > 2$. Combining Lemmas 12.4.6 and 12.4.8, we have a 2QCFA M' with $\text{markers}[M] = (m + (k + 1) - 2)$, and accepting $L_{k+1}(L)$. \square

It is important to note that as we move from L to $L_1(L)$, if a probabilistic automaton were used instead of a quantum one, the expected time would have been exponential. Replacing each x_i in an input $y = x_1\#x_2\#\dots\#x_{2n}$ by a if $x_i \in L$ and by b otherwise, the resulting language is still non-regular. Therefore, any 2PFA would require exponential time to accept this language [45]. This same reasoning holds for every level $k \geq 1$ in the M2PFA model. This result shows a quantum-classical complexity separation.

If $0\text{-sided}[2QCFA] = 1\text{-sided}[2QCFA]$ or $1\text{-sided}[2QCFA] = 2\text{-sided}[2QCFA]$, a simple induction using Lemma 12.4.9 or Lemma 12.4.10 would show that the language hierarchy $L_k(L)$ would collapse, in the sense that a fixed number of markers could be used to recognize $L_k(L)$ for every k , in the quantum model.

Theorem 12.4.12. *If M_0 with two-sided error ϵ can be amplified to an arbitrary error $\frac{\epsilon}{p(n)}$ for some polynomial $p(n)$, then $L_k(L) \in QH(\text{markers}[L])$, for every k .*

It is more likely that $0\text{-sided}[2QCFA] \subset 1\text{-sided}[2QCFA] \subset 2\text{-sided}[2QCFA]$ with strict inclusions. In this case, there would be a language L requiring a two-sided error

2QCFA M_0 to be recognized. We could use L to derive the language $L_1(L)$. For a word $y = x_1\#x_2\#\dots\#x_{2n}$, n could be made arbitrarily large, in which case M_0 's error would need to be amplified. If $\text{markers}[M_0] = 0$, we cannot even use a marker to store the x_i 's position. In all cases, the amplification needs to be done using only a constant number of classical and quantum states, otherwise extra markers would be required. As these obstacles seem hard to overcome in the 2QCFA model, we conjecture that this amplification is impossible. An important implication of this conjecture is a hierarchy of languages not recognized by 2QCFA. Currently, there is no known language which cannot be recognized by this model.

Conjecture 12.4.13. *Suppose $0\text{-sided}[2QCFA] \subset 1\text{-sided}[2QCFA] \subset 2\text{-sided}[2QCFA]$ with strict inclusions. Let $L \in 2\text{-sided}[2QCFA] \setminus 1\text{-sided}[2QCFA]$, and let M_0 be a 2QCFA recognizing L . Let the input word be $y = x_1\#x_2\#\dots\#x_{2n}$. If M_0 's error cannot be amplified, then $L_{k+1}(L) \notin QH(k + \text{markers}[L])$.*

In the classical paradigm, one or two markers are equivalent, that is, $L[1\text{-M2DFA}] = L[2\text{-M2DFA}]$ follows easily from [79]. Nonetheless, in the quantum case, a 1-M2QCFA can recognize languages not known to be recognizable by 2QCFA. An example is the language $L_{\text{middle}} = \{xay \mid x, y \in \{a, b\}^* \text{ and } |x| = |y|\}$.

For brevity, we give a sketch of the proof that L_{middle} is in $L[1\text{-M2QCFA}]$. It is known that we can get a one-sided error 2QCFA for L_{middle} . For an input $x = \sigma_1\sigma_2\dots\sigma_l$, it is possible to sequentially place a marker in each σ_i and test if $|\sigma_1\dots\sigma_{i-1}| = |\sigma_{i+1}\dots\sigma_l|$ using a simple variation of this automaton. From Lemma 12.5.1, this can be accomplished with high probability. In case the test succeeds, if $\sigma_i = a$ the input is accepted, otherwise it is rejected. A similar procedure is used to find the n -th symbol $\#$ in Lemma 12.4.8.

12.4.2 Computability Separation for Multi-head Finite Automata

In this section, we answer an open question raised in [103]. That is, we show that an m -H2QCFA can recognize languages that m -H2DFA cannot. Firstly, we prove that for positive integers k_1 and k_2 , with $k_1 > k_2$, a H2DFA for $L_{k_1}(L)$ requires more heads than a H2DFA that accepts $L_{k_2}(L)$. Then, the claim follows by a similar reasoning as was done for the multi-head case.

Theorem 12.4.14. *Let L be a language recognized by a non-sensing multi-head 2DFA. Then $L_{k+1}(L)$ requires more heads than $L_k(L)$.*

Proof. For a fixed m , suppose that there is a non-sensing m -H2DFA, M , for $L_k(L)$ and $L_{k+1}(L)$. It is trivial to see that M can be simulated by a m -M2DFA. It suffices to use one

marker for each head of M . We have two levels of the language hierarchy being recognized by the same number of markers which is a contradiction to Theorem 12.4.14. \square

Theorem 12.4.15. $L[k\text{-H2DFA}] \subset L[k\text{-H2QCFA}]$, with strict containment for $k \geq 3$.

Proof. The reasoning is analogous to the proof of Theorem 12.4.11. Assume L requires m heads to be accepted in the H2DFA model. From L to $L_1(L)$, the proof is the same. From $L_1(L)$ to $L_2(L)$, three heads will be needed: one to mark the x_i being amplified, other to run through the whole input to simulate a probability depending on n , and a last one to provide the sensing feature. If the number of heads required by L is greater than or equal to 3, no extra heads are necessary. From $L_k(L)$ to $L_{k+1}(L)$, we need the three heads of the previous cases, as well as one extra head to keep the gambler's ruin game status. \square

12.5 General Amplification Lemmas

In this section, we develop amplification lemmas that can be used in any 2QCFA. Lemmas 12.5.1 and 12.5.4 treat one-sided and two-sided 2QCFA amplification, respectively.

12.5.1 One-sided amplification

The following lemma states that it is possible to amplify a one-sided error 2QCFA to a polynomially small error.

Lemma 12.5.1 (One-sided amplification). *Let M be a 2QCFA, with one-sided error ϵ , that recognizes a language L , and let n denote some parameter. If it is possible to simulate a probability $p \leq \frac{\epsilon}{2f(n)\log f(n)}$, then it is possible to build a one-sided error 2QCFA M' accepting L with error at most $\frac{\epsilon}{f(n)}$. Moreover, M' simulates M an expected $O(\frac{1}{p})$ times.*

Proof. We exploit the fact that amplifying a one-sided error 2QCFA can be simply done by a probabilistic procedure. To achieve the desired error, it suffices to probabilistically simulate M enough times with high probability. If M rejects, we know with certainty that the input x is not in L .

Automaton M' is specified in Algorithm 23. Let n_{it} be the minimum number of iterations to achieve the desired error. Note that since M is a one-sided error 2QCFA, the error decreases exponentially with n_{it} . We want the error after n_{it} iterations to be:

$$\epsilon^{n_{it}} \leq \frac{\epsilon}{2f(n)} . \quad (12.10)$$

Then, n_{it} can be bounded below by:

$$n_{it} \geq \frac{1}{\log \epsilon} (\log \epsilon - \log 2 - \log f(n)) . \quad (12.11)$$

```

1 Let  $M$  be a 2QCFA with one-sided error  $\epsilon$ ;
2 Let  $p_{stop} = p$  be the probability of stopping the procedure ;
3 while  $Rand[(1 - p_{stop})]$  do
4   | Simulate  $M$  on  $x$  ;
5   | if  $M$  rejects  $x$  then
6   |   | Reject ;
7   | end
8 end
9 Accept ;

```

Algorithm 23: 2QCFA M' for one-sided error amplification

So, there is a suitable constant c' such that $n_{it} = c' \log f(n)$, and satisfying the previous equation. Let $(1 - p_{stop})$ be the probability of running M one more time. To run M n_{it} times with high probability, we have:

$$(1 - p_{stop})^{n_{it}} \geq 1 - \frac{\epsilon}{2f(n)} . \quad (12.12)$$

Using Bernoulli's inequality, that is $(1 + x)^n \geq 1 + nx$ for $x > -1$, we can obtain a lower bound for $(1 - p_{stop})^{n_{it}}$. We just have to make the lower bound greater than $1 - \frac{\epsilon}{2f(n)}$:

$$1 - n_{it}p_{stop} \geq 1 - \frac{\epsilon}{2f(n)} . \quad (12.13)$$

Then,

$$p_{stop} \leq \frac{\epsilon}{2f(n)n_{it}} . \quad (12.14)$$

The probability of running the desired number of iteration is at least $(1 - \frac{\epsilon}{2f(n)})$. In this case, the success probability is at least $(1 - \frac{\epsilon}{2f(n)})$. Therefore, the overall success probability is at least:

$$(1 - \frac{\epsilon}{2f(n)})(1 - \frac{\epsilon}{2f(n)}) \geq 1 - \frac{\epsilon}{f(n)} . \quad (12.15)$$

Denote the expected number times M will be simulated by $E[n_{it}]$. Then,

$$E[n_{it}] = p_{stop} \sum_{i \geq 1} i(1 - p_{stop})^i = p_{stop} \frac{(1 - p_{stop})}{(1 - (1 - p_{stop}))^2} = \frac{(1 - p_{stop})}{p_{stop}} . \quad (12.16)$$

Since $1 - p_{stop} > \frac{1}{2}$ for n sufficiently large, $E[n_{it}]$ is $O(\frac{1}{p_{stop}})$. □

Corollary 12.5.2. *If M runs in expected polynomial time, then M' also runs in expected polynomial time.*

Corollary 12.5.3. *Let M be a 2QCFA with one-sided error ϵ , and let n denote the input size. It is possible to build a 2QCFA, M' , with one-sided error $\frac{\epsilon}{f(n)}$, where $f(n)$ is a polynomial, or $f(n) = 2^n$.*

Proof. We know how to generate probabilities of the form $\frac{1}{n}$ using the input word [7]. Besides, we also know how to generate probabilities of the form $\frac{1}{2}$ using a Hadamard gate of one qubit. In order to generate a more sophisticated probability such as $\frac{1}{2^{in}}$, we run the first procedure j times, run the second procedure i times, and accept only if all succeed \square

12.5.2 Two-sided amplification

Lemma 12.5.4 states that is possible to amplify a one-sided error 2QCFA to a polynomially small error, given the required space to store the gambler's ruin game state.

Lemma 12.5.4 (Two-sided amplification). *Let M be a 2QCFA, with two-sided error ϵ , that recognizes a language L , and let n denote some parameter. If it is possible to store and update the state of a gambler's ruin game in which each player starts with n coins, then it is possible to build a two-sided error 2QCFA M' accepting L with error $\frac{\epsilon}{f(n)}$, where $f(n)$ is $\cup_{k>1} O(k^n)$. Moreover, the expected number of times that M' simulates M is $O(n)$.*

Proof. The amplification proof relies on simple results of the gambler's ruin problem. A gambler with $coins_g$ coins makes bets in a casino with $coins_c$ coins. The gambler wins each bet with probability p . The game ends only when the gambler or the casino is ruined, loosing all of their coins. This problem is described in more depth in [35].

In the automata case, each simulation of M on input x can be viewed as a bet with probability $p = 1 - \epsilon$. Define q as $q = 1 - p$. The construction simulates a simple gambler's ruin problem and is described in Algorithm 24. The probability of the gambler winning the game is:

$$P_{gambler} = \frac{1 - (\frac{q}{p})^n}{1 - (\frac{q}{p})^{2n}} . \quad (12.17)$$

Since $q < p$, we have:

$$P_{gambler} \geq 1 - (\frac{q}{p})^n . \quad (12.18)$$

The fraction $\frac{q}{p}$ is equal to $\frac{\epsilon}{1-\epsilon}$. From the definition of a M2QCFA, we have that $\frac{q}{p} < \frac{1}{k}$, for every constant $k > 1$. Therefore, the error becomes exponentially small in n .

The total success probability, p_{suc} , is:

$$p_{suc} \geq 1 - (\frac{1}{k})^n . \quad (12.19)$$

```

1 Let  $x$  be the input word ;
2 Let  $M$  be the 2QCFA to be amplified ;
3 Let  $coins_g$  be the number of gambler's coins ;
4 Let  $coins_c$  be the number of casino's coins ;
5  $coins_g = coins_c = n$  ;
6 while  $coins_g > 0$  and  $coins_c > 0$  do
7   | Run  $M$  on  $x$  ;
8   | if  $M$  accepts  $x$  then
9     |   Increment  $coins_g$  ;
10    |   Decrement  $coins_c$  ;
11  | end
12  | else
13    |   Decrement  $coins_g$  ;
14    |   Increment  $coins_c$  ;
15  | end
16 end
17 if  $coins_g = 2n$  then
18 |   Accept ;
19 end
20 else
21 |   Reject ;
22 end

```

Algorithm 24: 2QCFA M' for two-sided error amplification

For every k , and with $f(n)$ in $O(k^n)$, this probability is bounded below by:

$$p_{suc} \geq 1 - \frac{\epsilon}{f(n)} . \quad (12.20)$$

The expected number of times M will be simulated for a given x_i is given by the equation from gambler's ruin problem:

$$n_{it} = \frac{2n}{p-q} \frac{1 - (\frac{q}{p})^n}{1 - (\frac{q}{p})^{2n}} - \frac{2n}{2(p-q)} . \quad (12.21)$$

Since $p > q$, n_{it} can be bounded by:

$$n_{it} \leq \frac{2n}{p-q} . \quad (12.22)$$

From the definition of a 2QCFA, p and q are constants. Hence n_{it} is $O(n)$. Therefore, if M runs in polynomial time, M' will also run in polynomial time \square

12.6 Discussion

In this chapter, we introduced the M2QCFA model by allowing the 2QCFA model to use markers. For any number of markers, we showed that this new model is more powerful than its classical analogues M2DFA and M2PFA models in terms of computability and complexity, respectively.

Regarding the open question of whether the quantum k -H2QCFA model is more expressive than classical k -H2DFA model, we showed that the language $L_{pow(k)}$, which was proposed as an indication of their separation is, in fact, not a good witness. There is a 5-H2DFA recognizing this language, for any k . Nonetheless, we answered it affirmatively by extending our result for the M2QCFA.

The following is a list of important open questions concerning the computational power of the 2QCFA model and its variations.

- Is the set of languages recognized by a k -M2QCFA properly contained in the set of languages recognized by a $(k + 1)$ -M2QCFA?
- Given access to a biased coin whose probability p of turning heads is a function of the input length, can a 2QCFA check if $p > \frac{1}{2}$?
- Is it possible to improve the following containments in terms of the number of markers and heads:

$$L[k\text{-M2QCFA}] \subseteq L[(k+1)\text{-SH2QCFA}] \subseteq L[(k+2)\text{-H2QCFA}]?$$

- Is there a context free language that cannot be recognized by the 2QCFA model?

Chapter 13

Conclusion

Besides providing an extensive exposition to Quantum Computational Complexity and Quantum Finite Automata, this thesis brings four main contributions. Known results about $\text{QMA}(k)$ are derived in a unified way by establishing the interplay of classical PCP verifiers and this quantum class. Secondly, we show that some paths which can be used to improve the complexity bounds of $\text{QMA}(k)$ are obstructed by computational hardness. Variations of $\text{QMA}(k)$ are explored. Notably, building on known results, we show that increasing the flexibility of a $\text{QMA}(k)$ verifier by allowing a constant number of logarithmic sized queries to unentangled provers does not increase the power of this class. In Quantum Finite Automata, we show that a quantum variant of multi-marker and multi-head automata is more powerful than their classical analogues for any number of markers or heads.

There are several important open questions that remain to be answered. In Quantum Complexity, we can highlight the importance of finding non-trivial bounds for $\text{QMA}(k)$. Currently, the tightest known containments are $\text{QMA} \subseteq \text{QMA}(k) \subseteq \text{NEXP}$. Another important line of research is to improve the completeness soundness gap of $\text{QMA}(k)$ protocols for NP-complete languages. Any improvement automatically enhances our computational hardness results for disentanglers and de Finetti theorems. Finding a language that the quantum finite automaton 2QCFA can not recognize is a central open question. However, we stress that the definition of this model is very flexible allowing any unitary as long as it acts on a constant dimensional space. We wonder if this flexibility would allow the recognition of all languages.

Bibliography

- [1] Complexity zoo. <https://complexityzoo.uwaterloo.ca/>, 2015. [Online; accessed 24-Jan-2015].
- [2] AARONSON, S. The quantum pcg manifesto. <http://www.scottaaronson.com/blog/?p=139>, 2006. [Online; accessed 25-Aug-2015].
- [3] AARONSON, S., BEIGI, S., DRUCKER, A., FEFFERMAN, B., AND SHOR, P. W. The power of unentanglement. *Electronic Colloquium on Computational Complexity (ECCC)* 15, 051 (2008).
- [4] AARONSON, S., IMPAGLIAZZO, R., AND MOSHKOVITZ, D. AM with multiple merlins. In *IEEE 29th Conference on Computational Complexity, CCC 2014, Vancouver, BC, Canada, June 11-13, 2014* (2014), pp. 44–55.
- [5] AARONSON, S., AND KUPERBERG, G. Quantum versus classical proofs and advice. In *IEEE Conference on Computational Complexity* (2007), IEEE Computer Society, pp. 115–128.
- [6] AHARONOV, D., ARAD, I., AND VIDICK, T. Guest column: The quantum pcg conjecture. *SIGACT News* 44, 2 (June 2013), 47–79.
- [7] AMBAINIS, A., AND WATROUS, J. Two-way finite automata with quantum and classical states. *Theor. Comput. Sci.* 287, 1 (Sept. 2002), 299–311.
- [8] ARAD, I., KITAEV, A., LANDAU, Z., AND VAZIRANI, U. An area law and sub-exponential algorithm for 1d systems. *CoRR abs/1301.1162* (2013).
- [9] ARAD, I., LANDAU, Z., AND VAZIRANI, U. An improved 1d area law for frustration-free systems. *CoRR abs/1111.2970* (2011).
- [10] ARORA, S., AND BARAK, B. *Computational Complexity: A Modern Approach*, 1st ed. Cambridge University Press, New York, NY, USA, 2009.

- [11] ARORA, S., LUND, C., MOTWANI, R., SUDAN, M., AND SZEGEDY, M. Proof verification and the hardness of approximation problems. *J. ACM* 45, 3 (May 1998), 501–555.
- [12] ASHWIN, N., AND PETER, S. On bit-commitment based quantum coin flipping. *CoRR abs/quant-ph/0206123* (2002).
- [13] BARAK, B., MOITRA, A., O'DONNELL, R., RAGHAVENDRA, P., REGEV, O., STEURER, D., TREVISAN, L., VIJAYARAGHAVAN, A., WITMER, D., AND WRIGHT, J. Beating the random assignment on constraint satisfaction problems of bounded degree. *CoRR abs/1505.03424* (2015).
- [14] BEIGI, S. NP vs qma_log(2). *Quantum Information & Computation* 10, 1&2 (2010), 2.
- [15] BEIGI, S., SHOR, P., AND WATROUS, J. Quantum interactive proofs with short messages. *Theory of Computing* 7, 7 (2011), 101–117.
- [16] BENNETT, C. H., BRASSARD, G., CRÉPEAU, C., JOZSA, R., PERES, A., AND WOOTTERS, W. K. Teleporting an unknown quantum state via dual classical and einstein-podolsky-rosen channels. *Phys. Rev. Lett.* 70 (Mar 1993), 1895–1899.
- [17] BLIER, H., AND TAPP, A. All languages in np have very short quantum proofs. *Quantum, Nano, and Micro Technologies, First International Conference on 0* (2009), 34–37.
- [18] BRANDÃO, F., CHRISTANDL, M., AND YARD, J. Faithful squashed entanglement. *Communications in Mathematical Physics* 306, 3 (2011), 805–830.
- [19] BRANDÃO, F. G., AND HARROW, A. W. Quantum de finetti theorems under local measurements with applications. *STOC '13, ACM*, pp. 861–870.
- [20] BRANDÃO, F. G. S. L., CHRISTANDL, M., AND YARD, J. A quasipolynomial-time algorithm for the quantum separability problem. In *STOC* (2011), pp. 343–352.
- [21] CHAILLOUX, A., AND SATTATH, O. The complexity of the separable hamiltonian problem. In *IEEE Conference on Computational Complexity* (2012), IEEE, pp. 32–41.
- [22] CHEN, J., AND DRUCKER, A. Short multi-prover quantum proofs for sat without entangled measurements. *CoRR abs/1011.0716* (2010).

- [23] CHIESA, A., AND FORBES, M. A. Improved soundness for qma with multiple provers. *CoRR abs/1108.2098* (2011).
- [24] CHRISTANDL, M., KÖNIG, R., MITCHISON, G., AND RENNER, R. One-and-a-half quantum de finetti theorems. *Communications in Mathematical Physics* 273, 2 (2007), 473–498.
- [25] CLAUSER, J. F., HORNE, M. A., SHIMONY, A., AND HOLT, R. A. Proposed experiment to test local hidden-variable theories. *Phys. Rev. Lett.* 23 (Oct 1969), 880–884.
- [26] CONDON., A. *Bounded error probabilistic finite state automata, Handbook on Randomized Computing*. Kluwer, 2001.
- [27] DEUTSCH, D. Quantum theory, the Church-Turing principle and the universal quantum computer. *Proceedings of the Royal Society of London A* 400 (1985), 97–117.
- [28] DIACONIS, P., AND FREEDMAN, D. Finite exchangeable sequences. *Ann. Probab.* 8, 4 (08 1980), 745–764.
- [29] DINUR, I. The pcg theorem by gap amplification. *J. ACM* 54, 3 (June 2007).
- [30] DOHERTY, A. C., PARRILO, P. A., AND SPEDALIERI, F. M. Complete family of separability criteria. *Phys. Rev. A* 69 (Feb 2004), 022308.
- [31] DURIS, P., AND GALIL, Z. Sensing versus nonsensing automata. In *Automata, Languages and Programming*, vol. 944 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1995, pp. 455–463.
- [32] DWORK, C., AND STOCKMEYER, L. Finite state verifiers i: The power of interaction. *J. ACM* 39, 4 (Oct. 1992), 800–828.
- [33] EINSTEIN, A., PODOLSKY, B., AND ROSEN, N. Can quantum-mechanical description of physical reality be considered complete? *Phys. Rev.* 47 (May 1935), 777–780.
- [34] FEIGE, U., AND KILIAN, J. Making games short (extended abstract). In *Proceedings of the Twenty-ninth Annual ACM Symposium on Theory of Computing* (New York, NY, USA, 1997), STOC '97, ACM, pp. 506–516.
- [35] FELLER, W. *An Introduction to Probability Theory, Vol. 1*, third ed. Wiley, New York, NY, 1968.

- [36] FEYNMAN, R. P. Simulating physics with computers. *International Journal of Theoretical Physics* 21, 6-7 (1982), 467–488.
- [37] F.G.S.L. BRANDAO, A. H. Replacing hierarchies with nets, 2015.
- [38] FITZSIMONS, J., AND VIDICK, T. A multiprover interactive proof system for the local hamiltonian problem. In *Proceedings of the 2015 Conference on Innovations in Theoretical Computer Science* (New York, NY, USA, 2015), ITCS '15, ACM, pp. 103–112.
- [39] FUCHS, C. A., AND VAN DE GRAAF, J. Cryptographic distinguishability measures for quantum-mechanical states. *IEEE Transactions on Information Theory* 45, 4 (1999), 1216–1227.
- [40] GALL, F., NAKAGAWA, S., AND NISHIMURA, H. On qma protocols with two short quantum proofs. *Quantum Information and Computation* 12, 7-8 (2012), 589–600.
- [41] GAREY, M. R., AND JOHNSON, D. S. *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA, 1990.
- [42] GHARIBIAN, S. Strong np-hardness of the quantum separability problem. *Quantum Info. Comput.* 10, 3 (Mar. 2010), 343–360.
- [43] GHARIBIAN, S., SIKORA, J., AND UPADHYAY, S. QMA variants with polynomially many provers. *Quantum Information & Computation* 13, 1-2 (2013), 135–157.
- [44] GOLDWASSER, S., AND SIPSER, M. Private coins versus public coins in interactive proof systems. In *Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing* (New York, NY, USA, 1986), STOC '86, ACM, pp. 59–68.
- [45] GREENBERG, A. G., AND WEISS, A. A lower bound for probabilistic algorithms for finite state machines. *J. Comput. Syst. Sci.* 33, 1 (Aug. 1986), 88–105.
- [46] GRILO, A. B., AND MOURA, A. V. Language classes and quantum finite automata. In *Proceedings of the the IV Workshop-School in Quantum Computation and Information* (2012), WECIC '12, pp. 90 – 95.
- [47] GROVER, L. K. A fast quantum mechanical algorithm for database search. In *28th Annual ACM Symposium on the Theory of Computing* (1996), p. 212. New York.
- [48] GURUSWAMI, V. Query-efficient checking of proofs and improved pcg characterizations of np, 1999.

- [49] GUTOSKI, G., AND WATROUS, J. Toward a general theory of quantum games. In *Proceedings of the Thirty-ninth Annual ACM Symposium on Theory of Computing* (New York, NY, USA, 2007), STOC '07, ACM, pp. 565–574.
- [50] HARROW, A. W. The church of the symmetric subspace. *CoRR abs/1308.6595* (2013).
- [51] HARROW, A. W., AND MONTANARO, A. Testing product states, quantum merlin-arthur games and tensor optimization. *J. ACM* 60, 1 (2013), 3.
- [52] HOLZER, M., KUTRIB, M., AND MALCHER, A. Multi-head finite automata: Characterizations, concepts and open problems. In *CSP* (2008), T. Neary, D. Woods, A. K. Seda, and N. Murphy, Eds., vol. 1 of *EPTCS*, pp. 93–107.
- [53] HOPCROFT, J., AND ULLMAN, J. *Introduction to automata theory, languages, and computation*. Addison-Wesley series in computer science. Addison-Wesley, 1999.
- [54] HSIA, P., AND YEH, R. T. Finite automata with markers. In *ICALP* (1972), pp. 443–451.
- [55] IMPAGLIAZZO, R., AND PATURI, R. On the complexity of k-sat. *J. Comput. Syst. Sci.* 62, 2 (2001), 367–375.
- [56] IOANNOU, L. M. Computational complexity of the quantum separability problem. *Quantum Info. Comput.* 7, 4 (May 2007), 335–370.
- [57] ITO, T., AND VIDICK, T. A multi-prover interactive proof for nexp sound against entangled provers. In *Proceedings of the 2012 IEEE 53rd Annual Symposium on Foundations of Computer Science* (Washington, DC, USA, 2012), FOCS '12, IEEE Computer Society, pp. 243–252.
- [58] JAIN, R., JI, Z., UPADHYAY, S., AND WATROUS, J. QIP = PSPACE. In *Proceedings of the 42nd ACM Symposium on Theory of Computing, STOC 2010, Cambridge, Massachusetts, USA, 5-8 June 2010* (2010), pp. 573–582.
- [59] JI, Z. Classical verification of quantum proofs. *CoRR abs/1505.07432* (2015).
- [60] JORDAN, S. P., KOBAYASHI, H., NAGAJ, D., AND NISHIMURA, H. Achieving perfect completeness in classical-witness quantum merlin-arthur proof systems. *Quantum Information & Computation* 12, 5-6 (2012), 461–471.

- [61] KEMPE, J., KOBAYASHI, H., MATSUMOTO, K., TONER, B., AND VIDICK, T. Entangled games are hard to approximate. In *Proceedings of the 2008 49th Annual IEEE Symposium on Foundations of Computer Science* (Washington, DC, USA, 2008), FOCS '08, IEEE Computer Society, pp. 447–456.
- [62] KITAEV, A., AND WATROUS, J. Parallelization, amplification, and exponential time simulation of quantum interactive proof systems. In *Proceedings of the Thirty-Second Annual ACM Symposium on Theory of Computing, May 21-23, 2000, Portland, OR, USA* (2000), pp. 608–617.
- [63] KITAEV, A. Y., SHEN, A. H., AND VYALYI, M. N. *Classical and Quantum Computation*. American Mathematical Society, Boston, MA, USA, 2002.
- [64] KOBAYASHI, H., LE GALL, F., AND NISHIMURA, H. Stronger methods of making quantum interactive proofs perfectly complete. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science* (New York, NY, USA, 2013), ITCS '13, ACM, pp. 329–352.
- [65] KOBAYASHI, H., MATSUMOTO, K., AND YAMAKAMI, T. Quantum merlin-arthur proof systems: Are multiple merlins more helpful to arthur? In *ISAAC* (2003), T. Ibaraki, N. Katoh, and H. Ono, Eds., vol. 2906 of *Lecture Notes in Computer Science*, Springer, pp. 189–198.
- [66] KONDACS, A., AND WATROUS, J. On the power of quantum finite state automata. In *Proceedings of the 38th Annual Symposium on Foundations of Computer Science* (Washington, DC, USA, 1997), IEEE Computer Society, pp. 66–.
- [67] LANDAU, Z., VAZIRANI, U., AND VIDICK, T. An efficient algorithm for finding the ground state of 1d gapped local hamiltonians. In *Proceedings of the 5th Conference on Innovations in Theoretical Computer Science* (New York, NY, USA, 2014), ITCS '14, ACM, pp. 301–302.
- [68] LI, K., AND SMITH, G. Quantum de finetti theorem measured with fully one-way locc norm. *CoRR abs/1408.6829* (2014).
- [69] MARRIOTT, C., AND WATROUS, J. Quantum arthur-merlin games. *Computational Complexity* 14, 2 (2005), 122–152.
- [70] MEIR, O. Combinatorial pcps with efficient verifiers. In *FOCS* (2009), IEEE Computer Society, pp. 463–471.

- [71] MITZENMACHER, M., AND UPFAL, E. *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, New York, NY, USA, 2005.
- [72] MOORE, C., AND CRUTCHFIELD, J. P. Quantum automata and quantum grammars. *Theor. Comput. Sci.* 237 (April 2000), 275–306.
- [73] MOSHKOVITZ, D., AND RAZ, R. Two query pcp with sub-constant error. *Electronic Colloquium on Computational Complexity (ECCC)* 15, 071 (2008).
- [74] NIELSEN, M., AND CHUANG, I. *Quantum computation and quantum information*. Cambridge Series on Information and the Natural Sciences. Cambridge University Press, 2000.
- [75] PAPADIMITRIOU, C. M. *Computational complexity*. Addison-Wesley, Reading, Massachusetts, 1994.
- [76] PERESZLÉNYI, A. On quantum interactive proofs with short messages. *CoRR abs/1109.0964* (2011).
- [77] PERESZLÉNYI, A. Multi-prover quantum merlin-arthur proof systems with small gap. *CoRR abs/1205.2761* (2012).
- [78] PETERSEN, H. Automata with sensing heads. In *Theory of Computing and Systems, 1995. Proceedings., Third Israel Symposium on the* (Jan 1995), pp. 150–157.
- [79] PETERSEN, H. The equivalence of pebbles and sensing heads for finite automata. In *Fundamentals of Computation Theory*, B. Chlebus and L. Czaja, Eds., vol. 1279 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 1997, pp. 400–410.
- [80] PRESKILL, J. Quantum computing and the entanglement frontier. *CoRR abs/1203.5813* (2012).
- [81] QIU, D. Some observations on two-way finite automata with quantum and classical states. In *Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues*, D.-S. Huang, I. Wunsch, Donald C., D. Levine, and K.-H. Jo, Eds., vol. 5226 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2008, pp. 1–8.
- [82] QIU, D. Some observations on two-way finite automata with quantum and classical states. In *Advanced Intelligent Computing Theories and Applications. With Aspects of Theoretical and Methodological Issues*, D.-S. Huang, I. Wunsch, Donald C.,

- D. Levine, and K.-H. Jo, Eds., vol. 5226 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, 2008, pp. 1–8.
- [83] RAZ, R. Quantum information and the PCP theorem. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS 2005), 23-25 October 2005, Pittsburgh, PA, USA, Proceedings* (2005), pp. 459–468.
- [84] REICHARDT, B. W., UNGER, F., AND VAZIRANI, U. A classical leash for a quantum system: Command of quantum systems via rigidity of chsh games. In *Proceedings of the 4th Conference on Innovations in Theoretical Computer Science* (New York, NY, USA, 2013), ITCS '13, ACM, pp. 321–322.
- [85] ROSSMAN, B., SERVEDIO, R. A., AND TAN, L. An average-case depth hierarchy theorem for boolean circuits. *Electronic Colloquium on Computational Complexity (ECCC) 22* (2015), 65.
- [86] ROWE, M. A., KIELPINSKI, D., MEYER, V., SACKETT, C. A., ITANO, W. M., MONROE, C., AND WINELAND, D. J. Experimental violation of a bell's inequality with efficient detection. *Nature 409* (2001), 791–794.
- [87] SAHAI, A., AND VADHAN, S. A complete problem for statistical zero knowledge. *J. ACM 50*, 2 (Mar. 2003), 196–249.
- [88] SAKURAI, J. J., AND NAPOLITANO, J. *Modern Quantum Mechanics*. Addison-Wesley, 2010.
- [89] SHAMIR, A. $\text{Ip} = \text{pspace}$. *J. ACM 39*, 4 (Oct. 1992), 869–877.
- [90] SHI, Y., AND WU, X. Epsilon-net method for optimizations over separable states. In *Proceedings of the 39th International Colloquium Conference on Automata, Languages, and Programming - Volume Part I* (2012), ICALP'12, Springer-Verlag, pp. 798–809.
- [91] SHOR, P. W. Algorithms for quantum computation: discrete logarithms and factoring. *Proceedings 35th Annual Symposium on Foundations of Computer Science 35* (1994), 124–134.
- [92] SIPSER, M. *Introduction to the Theory of Computation*. Thomson Course Technology, 2006.
- [93] TREVISAN, L. On the complexity of k-sat. *Bull. Amer. Math. Soc. 49* (2012), 91–111.

- [94] VANDENBERGHE, L., AND BOYD, S. Semidefinite programming. *SIAM Rev.* 38, 1 (Mar. 1996), 49–95.
- [95] VIDICK, T. Around the quantum pcg conjecture. http://users.cms.caltech.edu/~vidick/teaching/286_qPCP/index.html, 2014. [Online; accessed 25-Aug-2015].
- [96] WATROUS, J. Quantum computational complexity. In *Encyclopedia of Complexity and Systems Science*, R. A. Meyers, Ed. Springer, 2009, pp. 7174–7201.
- [97] WATROUS, J. Theory of quantum information. <https://cs.uwaterloo.ca/~watrous/LectureNotes.html>, 2011. [Online; accessed 25-Aug-2015].
- [98] WILDE, M. M. *Quantum Information Theory*, 1st ed. Cambridge University Press, New York, NY, USA, 2013.
- [99] YAKARYILMAZ, A. One-counter verifiers for decidable languages. *CoRR abs/1207.3880* (2012).
- [100] YAKARYILMAZ, A. Log-space counter is useful for unary languages by help of a constant-size quantum register. *CoRR abs/1309.4767* (2013).
- [101] YAO, A. C. Quantum circuit complexity. In *34th Annual Symposium on Foundations of Computer Science, 3-5 November 1993, Palo Alto, California, USA* (1993), IEEE, pp. 352–361.
- [102] YAO, A. C., AND RIVEST, R. L. $K + 1$ heads are better than k . *J. ACM* 25, 2 (Apr. 1978), 337–340.
- [103] ZHENG, S., LI, L., AND QIU, D. Two-tape finite automata with quantum and classical states. *International Journal of Theoretical Physics* 50, 4 (2011), 1262–1281.

Appendix A

Auxiliary Proofs

A.1 Mathematical Lemmas

We present a simple mathematical lemma used in the proof of 8.4.2. This lemma states that if it is possible to control the lower bound of the inner product of two unity length L_1 norm vectors, then there is a component in both vectors that can be made arbitrarily close to 1. This result is useful when enforcing that the value of a position in the quantum proof is consistent with respect to other proofs.

Lemma A.1.1. *Let $x = [x_1, \dots, x_n]$ and $y = [y_1, \dots, y_n]$ be vectors in \mathbb{R}^n where $x_i, y_i \geq 0$, for all i , and assume $\|x\|_1 = 1$ and $\|y\|_1 = 1$. Let $\langle x|y \rangle \geq k_1 > \frac{1}{2}$. For every $k_2 \in]0.5, 1[$, by adjusting k_1 it is possible to obtain a $j \in [n]$ such that $|x_j|$ and $|y_j|$ are greater than k_2 .*

Proof. Let θ be the smallest angle between x and y . From the hypothesis, their inner product is at least k_1 resulting in

$$\cos \theta \|x\|_2 \|y\|_2 = \langle x|y \rangle \geq k_1.$$

Without loss generality, assume that $\|x\|_2$ is greater than $\|y\|_2$. Then we have

$$\|x\|_2^2 \geq \cos \theta \|x\|_2 \|y\|_2 = \langle x|y \rangle \geq k_1.$$

Let x_1 be the greatest component of x . Since each component of x is non negative and $\|x\|_1 = 1$, the following inequality holds

$$(1 - x_1)^2 = \left(\sum_{i=2}^n x_i \right)^2 \geq \sum_{i=2}^n x_i^2.$$

Then, $\|x\|_2^2$ can be upper bounded by:

$$x_1^2 + (1 - x_1)^2 \geq \|x\|_2^2 \geq \cos \theta \|x\|_2 \|y\|_2 = \langle x|y \rangle \geq k_1.$$

Solving this simple quadratic equation yields the roots $\frac{2 \pm \sqrt{4 - 8(1 - k_1)}}{4}$. For $k_1 \in]0.5, 1[$, we note that the minimum value of $x_1 \geq 0$ satisfying the previous inequality is monotonically increasing in k_1 and varies in the range $]0.5, 1[$ as desired.

For vector y , we have

$$\|y\|_2 \geq \cos \theta \|x\|_2 \|y\|_2 = \langle x|y \rangle \geq k_1,$$

and a similar reasoning applies by squaring both sides. Moreover, vectors x and y can be made arbitrarily close to parallel ones since $k_1 \leq \cos \theta \|x\|_2 \|y\|_2 \leq \cos \theta$. Therefore, the same greatest component of x and y can be made arbitrarily large by controlling k_1 .

Note that this Lemma could also have been proved using the Hölder inequality, which states that

$$|\langle x|y \rangle| \leq \|x\|_p \|y\|_q$$

where p and q satisfy $\frac{1}{p} + \frac{1}{q} = 1$. By taking $p = 1$ and q unbounded, it is easy to see that the result follows. \square

If we intend to work with qubits instead of qudits, there is one simple technicality that we need to address. The PCP witness and the alphabet size should be a power of two. The next claim shows how to make the appropriate conversion.

Claim A.1.2. *Let V be a $\text{PCP}_{1,1-\epsilon_0}(r(n), q(n))_\Sigma$ verifier. It is possible to convert V to a $\text{PCP}_{1,1-\epsilon_0}(r(n), q(n))_{\Sigma'}$ verifier in which the new witness size m' and the new alphabet size $|\Sigma'|$ are a power of two. Further, if m is the witness size of V , then $m' \leq 2m$ and $|\Sigma'| \leq 2|\Sigma|$.*

Proof. It is possible to add positions to the witness $y = y_1 \dots y_m$ until its size becomes a power of two. Note that this process at most doubles its size. A similar reasoning applies to the original alphabet Σ .

The verifier can always ignore extra positions in the proof by not querying them, thus $r(n)$, $q(n)$ and the completeness remain the same. Also, if the new verifier reads a symbol that was not in the alphabet Σ it can readily reject. Therefore, soundness also remains the same. \square

Lemma A.1.3 (Cauchy-Schwarz). *Let v, w be two vectors in an inner product vector space V . It holds that*

$$|\langle v, w \rangle| \leq \sqrt{\langle v, v \rangle} \sqrt{\langle w, w \rangle}$$

where $\langle v, w \rangle$ is the inner product of v and w .

Proof. If $v = 0$ or $w = 0$, the lemma trivially holds. Suppose $v \neq 0$ and $w \neq 0$. We write v as the sum of two orthogonal vectors w and w^\perp where the latter we take to be

$$w^\perp = v - \frac{\langle w, v \rangle}{\langle w, w \rangle} w.$$

With these vectors, v becomes

$$v = \frac{\langle w, v \rangle}{\langle w, w \rangle} w + w^\perp.$$

Using the Pythagorean Theorem results in

$$\langle v, v \rangle = \left\langle \frac{\langle w, v \rangle}{\langle w, w \rangle} w, \frac{\langle w, v \rangle}{\langle w, w \rangle} w \right\rangle + \langle w^\perp, w^\perp \rangle.$$

Discarding the term $\langle w^\perp, w^\perp \rangle$ which is non-negative, we conclude that

$$\langle v, v \rangle \langle w, w \rangle \geq |\langle v, w \rangle|^2.$$

□